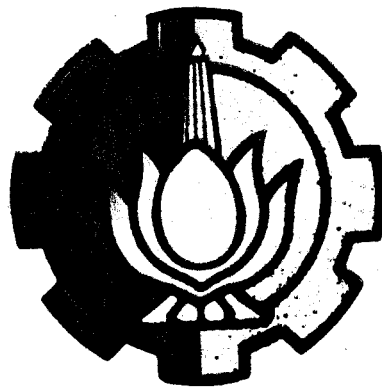


6947/ITS/H/95

**ALAT PENGUKUR KEKUATAN GIGIT
PADA PEMBUATAN
GIGI TIRUAN LENGKAP
YANG DAPAT DIINTERFACEKAN PADA PC**

PERPUSTAKAAN ITS	
Tgl. Terima	04 MAY 1994
Terima Dari	TA
No. Agenda Prp.	2066/B



RSE
621 398 1
Dya
d-1
1994

OLEH :

DYAH PATRIA N

NRP. 2882200974

**JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
1994**

**ALAT PENGUKUR KEKUATAN GIGIT
PADA PEMBUATAN
GIGI TIRUAN LENGKAP
YANG DAPAT DIINTERFACEKAN PADA PC**

TUGAS AKHIR

**Diajukan guna memenuhi sebagian
persyaratan untuk memperoleh
gelar Sarjana Teknik Elektro
pada
Bidang Studi Elektronika
Jurusan Teknik Elektro
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
Surabaya**

**Mengetahui / Menyetujui
Dosen Pembimbing**



**Ir. NAWANTOWIBOWO
NIP. 130368612**

**S U R A B A Y A
FEBRUARI, 1994**

ABSTRAK

Gigi merupakan salah satu alat yang vital bagi manusia. Gigi, pusat syaraf dan otot-otot penunjangnya memiliki kerjasama yang sangat baik dalam aktifitas-aktifitas yang dilakukan mulut seperti berbicara, menggigit, mengunyah ataupun menguap.

Seorang dengan gigi yang sehat kekuatan gigit antara kedua bagian rahang (rahang atas dan bawah) yang cukup tinggi. Perubahan kekuatan gigit ini terjadi jika ada perubahan pada gigi ataupun syaraf serta otot penunjangnya. Seorang pemakai gigi tiruan lengkap, kekuatan gigitnya hanya sepertiga dibandingkan saat masih memiliki gigi aslinya.

Alat pengukur kekuatan gigit ini dapat digunakan oleh seorang dokter gigi untuk mengetahui kekuatan gigi seorang pasien serta membantu menganalisa kemungkinan-kemungkinan adanya kerusakan pada gigi atau penunjangnya.

Alat ukur ini menggunakan sensor strain gage yang dirangkai pada jembatan wheatstone. Alat ini dijalankan dan dikontrol menggunakan minimum sistem 8088 serta dilengkapi dengan komunikasi serial pada PC untuk menampilkan grafik kekuatan gigit sebagai fungsi waktu.

KATA PENGANTAR

Puji syukur kami panjatkan kehadiran Allah swt atas ridlo dan rahmat yang dilimpahkanNya, sehingga penyusunan tugas Akhir yang berjudul " **Alat Pengukur Kekuatan Rahang pada Pemasangan Gigi Tiruan Lengkap yang dapat diinterfacekan pada PC** " ini dapat terselesaikan.

Tugas Akhir ini merupakan salah satu syarat untuk menyelesaikan program studi Strata satu (S1) pada jurusan Teknik Elektro, Fakultas Teknologi Industri, Institut Teknologi Sepuluh November Surabaya.

Penyelesaian Tugas Akhir ini dilakukan dengan merencanakan dan membuat alat berdasarkan teori yang didapat dari bangku kuliah, literatur, bimbingan dari dosen pembimbing serta pihak lain yang telah memberikan bantuan dan dorongan.

Pada kesempatan ini penulis ingin mengucapkan terima kasih yang tulus atas kesempatan, bantuan serta bimbingan yang telah penulis terima, kepada :

1. Dr Ir Moch Salehduddin selaku ketua Jurusan Teknik Elektro FTI ITS.
2. Ir Soetikno selaku Koordinator Bidang Studi Elektronika.
3. Dr.Ir.Handayani Tj, yang telah memberikan permasalahan untuk tugas akhir ini.
4. Ir. Nawantowibowo, selaku dosen pembimbing.

5. Prof.dr. Iskandar, selaku kepala Laboratorium Gigi Tiruan Lengkap Fakultas Kedokteran Gigi Universitas Airlangga Surabaya.

6. Ir Djoko Purwanto, selaku dosen wali.

7. Semua pihak yang banyak membantu sehingga terselesaikannya Tugas Akhir ini dengan baik.

Penulis menyadari adanya kekurangan dalam pembuatan Tugas Akhir ini oleh karena itu kritik dan saran dari berbagai pihak sangat diharapkan. Besar harapan penulis buku dan alat ini berguna bagi yang membutuhkan.

Surabaya, Januari 1994

Penyusun

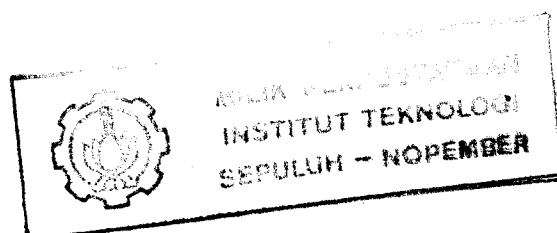
DAFTAR ISI

JUDUL	i
LEMBAR PENGESAHAN	ii
ABSTRAK	iii
KATA PENGANTAR	iv
DAFTAR ISI	vi
DAFTAR TABEL	x
DAFTAR GAMBAR	xi
 Bab I. Pendahuluan	 1
1.1 Latar Belakang	1
1.2 Permasalahan	2
1.3 Tujuan	2
1.4 Batasan Masalah	2
1.5 Metodologi	2
1.6 Sistematika Pembahasan	3
1.7 Relevansi	4
 Bab II Teori Penunjang	 5
2.1 Gigi dan Penunjangnya	5
2.2 Pengukur Regangan dan Pengukur	
Perubahan Resistansi yang Kecil	8
2.2.1 Pendahuluan untuk Pengaruh	
Regangan	8

2.2.2	Penggunaan Data Pengukur	
	Regangan	9
2.2.3	Pemasangan Pengukur Regangan	10
2.2.4	Pengukur Perubahan Resistansi	
	Kecil	11
	2.2.4.1 Jembatan Resistansi	
	Dasar	11
2.2.5	Efek-efek Panas dan Keseimbangan	
	Jembatan	13
2.2.6	Menyeimbangkan Sebuah Jembatan	
	Pengukur Regangan	14
2.3	Amplifier	15
	2.3.1 Op Amplifier Ideal	15
	2.3.2 Instrumentasi Amplifier	16
2.4	Analog to Digital (ADC)	17
	2.4.1 ADC Successive Aproximation	18
2.5	Minimum System μ P 8088	19
	2.5.1 Internal Register μ P 8088	21
	2.5.2 Basic Timing μ P 8088	22
	2.5.3 Read Only Memory (ROM)	22
	2.5.4 Random Acces Memory (RAM)	22
2.6	Decoder	24
2.7	Programmable Interval Timer 8253	25
2.8	Programmable Peripheral Interface 8255	26
2.9	Priority Interrupt Controller 8259	29

2.10	Prinsip Komunikasi Data Serial	35
2.10.1	Standart Komunikasi RS-232	36
2.10.2	Hubungan RS-232	37
2.10.3	Port Komunikasi Asinkron pada IBM PC	37
2.10.4	Inisialisasi Port Komunikasi	40
2.10.5	Membaca Data dari Port Komunikasi .	40
2.10.6	Mengirim Data melalui Port Komunikasi	40
Bab III	Perencanaan Hardware dan Software	41
3.1	Sensor pada Alat Ukur	41
3.1.1	Plat dan Tranducer	41
3.1.2	Rangkaian jembatan Wheatstone	43
3.2	Pengolah Sinyal Analog	45
3.2.1	Instrumentasi Amplifier	45
3.2.2	Filter	47
3.2.3	Multiplexer 4052	48
3.3	Pengolah Sinyal Digital	49
3.3.1	Konversi Analog ke Digital dengan ADC 0804	50
3.3.2	Central Processing Unit μ P 8088	53
3.3.3	Rangkaian Timer dan Interrupt	54

3.3.4	Programmable Peripheral Interface ..	56
3.3.5	Peraga (Display & Tombol)	56
3.4	Serial Komunikasi	60
Bab IV	Kalibrasi dan Pengukuran	63
4.1	Kalibrasi	63
4.2	Pengukuran	64
4.2.1	Output Instrumentasi Amplifier	64
4.2.2	Output Hasil Pembebanan	66
4.2.3	Hasil Pengukuran ADC	69
Bab V	Penutup	73
5.1	Kesimpulan	73
5.2	Saran-saran	74
Daftar Pustaka	75
Lampiran		



DAFTAR TABEL

Tabel 2.1.	Pemilihan Alamat Pada PIT 8253	26
Tabel 2.2.	Lokasi & Fungsi Register Pada PPI 8255	29
Tabel 2.3.	Pin-pin pada RS-232	37
Tabel 2.4.	Alamat Interface Serial Komunikasi	38
Tabel 2.5.	Line Control Register	39
Tabel 2.6.	Line Status Register	39
Tabel 4.1	Hasil Pengukuran Penguat Instrumentasi untuk pengukuran gigiseri	64
Tabel 4.2	Hasil Pengukuran Penguat Instrumentasi untuk Pengukuran gigi geraham kanan	65
Tabel 4.1	Hasil Pengukuran Penguat Instrumentasi untuk Pengukuran gigi geraham kiri	65

DAFTAR GAMBAR

Gambar 2.1.1. Asas Pergeseran dan Tantanan	
Pengunyahan	5
Gambar 2.2.1. Pengukur Regangan untuk Mengukur	
Perubahan Panjang Struktur	9
Gambar 2.2.2. Rangkaian Jembatan Wheatstone	
sederhana	11
Gambar 2.2.3 Pengukur Kerja dan Kompensasi Suhu	13
Gambar 2.2.4 Teknik Jembatan Wheatstone dengan	
Penyeimbang	14
Gambar 2.3.1 Rangkaian Dasar Instrumentasi Amplifier .	17
Gambar 2.4.1 Diagram Blok ADC Succesive Aproximation .	19
Gambar 2.5.1 Konfigurasi pin μ P 8088	21
Gambar 2.5.2 Basic Timing μ P 8088	23
Gambar 2.7.1 Internal Blok Diagram PIT 8253	25
Gambar 2.7.2 Format Control Word PIT 8253	27
Gambar 2.8.1 Internal Blok Diagram PPI 8255	28
Gambar 2.8.2 Format Control Word PPI 8255	30
Gambar 2.9.1 Internal Blok Diagram PIC 8259	31
Gambar 2.9.2 Format Initialization Command Word	33
Gambar 2.9.3 Format Operational Command Word	34
Gambar 2.10.1 Format karakter pada komp Asinkron	36
Gambar 3.1 Blok Diagram Perencanaan Hardware	41
Gambar 3.1.1 Bentuk Alat ukur	43
Gambar 3.1.2 Rangkaian Jembatan Wheatstone	44

Gambar 3.2.1	Blok Diagram Pengolah Sinyal Analog	44
Gambar 3.2.2	Operating Connection AD521JD	45
Gambar 3.2.3	Low Pass Filter	46
Gambar 3.2.4	Multiplexer 4052	47
Gambar 3.3.1	Blok Diagram Pengolah Sinyal Digital	49
Gambar 3.3.2	Rangkaian ADC 0804	49
Gambar 3.3.3	Timing Diagram ADC 0804	50
Gambar 3.3.4	Penghasil Tegangan Referensi 2,5V	51
Gambar 3.3.5	Memory Map	52
Gambar 3.3.5	Timing Diagram LCD	58

BAB I

PENDAHULUAN

I.1 LATAR BELAKANG

Gigi merupakan salah satu organ tubuh yang sangat penting bagi manusia. Manusia memerlukan energi untuk melaksanakan segala aktifitasnya, dan untuk mendapatkan energi ini salah satu cara yang utama adalah melalui makanan-makanan. Proses pencernaan makanan-makanan ini diawali dengan pengunyahan, gigi merupakan satu-satunya alat pengunyah bagi manusia. Dari sini terlihat betapa penting dan bergunanya gigi kita ini. Dari sisi lain dapat pula dilihat bahwa syaraf-syaraf gigi terhubung dengan bermacam-macam alat vital bagi manusia, sehingga kerusakan pada gigi yang kemudian berpengaruh pada syaraf gigi dapat mengakibatkan terkena pula penyakit-penyakit lain. Kesehatan gigi adalah hal yang sangat perlu diperhatikan oleh setiap insan. Pada gigi yang sehat kita dapat memperoleh kekuatan kunyah yang maksimum, berkurangnya kekuatan gigit dan kunyah seseorang disebabkan oleh perubahan yang ada pada gigi kita, misalnya kerusakan mahkota gigi, terkenanya syaraf gigi oleh penyakit ataupun usia yang semakin lanjut. Ada kalanya kita tidak menyadari adanya perubahan-perubahan pada gigi kita, dengan mengetahui kekuatan gigit atau kunyah pada saat tertentu seorang dokter gigi dapat menganalisa lebih lanjut apa yang menyebabkannya.

Jadi dengan adanya alat ini diharapkan para peneliti bidang Kedokteran Gigi dapat mempermudah pekerjaannya.

I.2 PERMASALAHAN

Dalam Tugas Akhir ini saya mencoba membuat alat berupa Alat Pengukuran Kekuatan Gigit pada pemasangan Gigi Tiruan Lengkap yang dapat diinterfacekan pada Personal Computer. dengan mengetahui kekuatan gigit atau kunyah dari pasien, seorang dokter gigi dapat memakai alat ukur ini sebagai pedoman pembuatan gigi tiruan (denture). Alat ini diharapkan juga dapat membantu para peneliti dibidang Kedokteran Gigi.

I.3 TUJUAN

Tujuan pembuatan tugas akhir ini adalah untuk mengukur kekuatan gigit pada manusia dan melihat respon dari pasien selama pengukuran.

I.4 BATASAN MASALAH

Dalam tugas akhir ini kekuatan gigit yang diukur adalah pada gigi seri dan kedua sisi gigi geraham. Pada gigi seri tumpuannya adalah pada gigi terdepan, sedangkan pada gigi geraham tumpuan dapat pada salah satu dari ketiga gigi geraham.

Hasil pengukuran ditampilkan pada display (LCD) serta dalam bentuk grafik pada layar monitor PC

I.5 METODOLOGI

Langkah-langkah penyusunan tugas akhir ini adalah sebagai berikut:

Mengumpulkan dan mempelajari literatur yang berhubungan dengan masalah yang dihadapi. Literatur ini adalah mengenai gigi dan otot-otot yang bekerja padanya, strain gage, instrumentasi amplifier, konversi analog ke digital, komunikasi data serial dan perangkat pendukung interface.

Merencanakan perangkat keras yang meliputi merencanakan rangkaian Wheatstone dengan penguat instrumentasi amplifier AD521JD, rangkaian pengolah sinyal analog ke digital serta penggunaan minimum sistem 8088 untuk mengendalikan rangkaian analog tersebut diatas. Grafik hasil pengukuran ditampilkan pada komputer IBM PC\XT.

Membuat perangkat keras serta perangkat lunak sesuai dengan perencanaan yang telah dibuat. Dilanjutkan dengan pengujian alat.

Menyimpulkan perancangan dan pembuatan alat sesuai dengan hasil pengujian. Memberikan saran-saran pengembangan alat lebih lanjut. Dari langkah-langkah diatas disusun buku laporan tugas akhir ini.

I.6 SISTEMATIKA PEMBAHASAN

Sistematika dari laporan tugas akhir ini disusun sebagai berikut:

Bab I adalah pendahuluan yang berisi tentang latar belakang, permasalahan, pembatasan masalah, tujuan, metodolo-

gi, sistematika dan relevansi.

Bab II adalah teori penunjang membahas mengenai, strain gage, pengolah sinyal analog, pengubah sinyal analog ke digital, minimum sistem 8088 serta PPI 8255.

Bab III adalah perencanaan perangkat keras yang meliputi perencanaan jembatan wheatstone beserta penyeimbangannya, amplifier, konversi analog ke digital, minimum sistem 8088 serta PPI 8255, serta perangkat lunak yang sesuai dengan perangkat keras tsb.

Bab IV adalah pengukuran dan pengujian alat ukur kekuatan gigit yang telah dirancang.

Bab V adalah penutup yang berisi kesimpulan dan saran pengembangan.

I.7 RELEVANSI

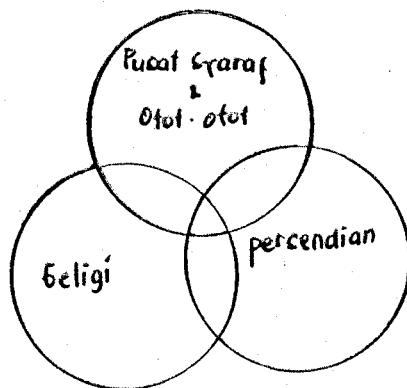
Hasil tugas akhir ini diharapkan dapat berguna pada laboratorium Kedokteran Gigi sebagai alat penunjang penelitian untuk pengembangan ilmu kedokteran gigi lebih lanjut.

BAB II

TEORI PENUNJANG

2.1. GIGI DAN PENUNJANGNYA

Sesungguhnya yang dimaksud dengan kekuatan rahang maupun kekuatan gigit atau kunyah merupakan kekuatan dari otot-otot yang bekerja untuk itu. Pergerakan rahang bawah dihasilkan oleh persitidakan otot pengunyahan, otot leher dan otot dada serta otot wajah. Yang dimaksud persitidakan adalah perpaduan fungsi dalam suatu rangkaian gerakan.



Gambar 2.1.1.

Asas Pergerakan tatanan pengunyah

Geligi menghasilkan hubungan *vertikal* (gerakan rahang dari atas ke bawah) dan *distal* (gerakan mendatar atau menggerus) rahang bawah yang mantap terhadap rahang atas karena hubungan yang baik antar mahkota gigi *setentanganya* (mahkota gigi yang berhadapan). Geligi juga menghasilkan bidang pemandu gerakan rahang bawah ke *anterior* (sebelah muka) dan

lateral (arah samping) pada deretan gerakan rahang bawah selama geligi bersentuhan. Bila kedua fungsi sendi ini serasi dengan geligi, terjadi *fungsi persarafototan* (kerjasama antara saraf dan otot) paling baik. Dan bila fungsi tatanan pengunyahan sehat, otot pengunyahan berfungsi serasi dan teratur. Keadaan ini memungkinkan gerakan disengaja dan tak disengaja rahang bawah menuju batas akhir gerakan. Didalam deretan gerakan ini terjadi gerakan berfungsi pengunyahan, penelanan, pengucapan dan pengkuapan yang wajar¹.

Variabel yang umumnya paling mengganggu keserasian fungsi tatanan adalah geligi dan persarafototan. Hubungan antar tonjol mahkota gigi pada hubungan mantap *maksilomandibular* (rahang atas-bawah) vertikal dan horisontal merupakan prasyarat keserasian fungsi. Sedang variabel kedua yaitu persarafototan yang dapat dipengaruhi oleh tanggapan kejiwaan. Keadaan ini menyebabkan tegangan otot yang berlebihan, merubah pola gerakan tidak disengaja atau kebiasaan merusak bukan fungsi. Pada umumnya tatanan pengunyahan berfungsi serasi dengan adanya derajat stress dan ketidakserasian oklusal tertentu.

Orang yang senantiasa merawat giginya secara teratur akan memiliki gigi yang sehat. Pada gigi yang sehat, kita memiliki kekuatan gigit serta kunyah yang maksimum/terbaik. Dari hasil pengukuran ternyata kekuatan mengunyah untuk molar pertama sekitar 100 lbs (\pm 50 kg) pada pria dan pada wa-

nita duapertiganya. Molar (gigi geraham) pertama adalah yang paling kuat, sedangkan premolar dan gigi depan hanya sepertiga gigi molar pertama, hal ini disebabkan karena perbedaan luas oklusal dan posisi insertationnya terhadap otot-otot pengunyah².

Pemakaian gigi tiruan (denture) lengkap ataupun sebagian juga menyebabkan berkurangnya kekuatan gigit kita dibandingkan jika kita menggunakan gigi asli kita. Hal ini disebabkan kerjasama antara gigi tiruan dengan otot pada rahang tidak sebaik apabila kita menggunakan gigi asli, atau bisa dikatakan gigi tiruan tidak akan bisa tertanam kuat pada rahang sebagaimana gigi asli kita. Apabila kita menggunakan Gigi Tiruan Lengkap kekuatan gigit kita hanya sepertiga kekuatan mengunyah gigi aslinya.

Faktor yang mempengaruhi kekuatan mengunyah adalah otot-otot mengunyah (yaitu otot-otot rahang, otot-otot lidah bagian atas dan bawah) dan kepekaan membran *periodontal* (jaringan termasuk gusi dan jaringan penyangga gigi) terhadap sakit. Yang terakhir ini lebih penting karena tekanan pengunyahan akan menekan membran *periodontal*.

Kemudian pada pemasangan Gigi Tiruan Lengkap itu sendiri kekuatan gigit kita berkurang lagi apabila pembuatannya kurang tepat bagi si pemakai, misalnya buah gigi yang terlalu tinggi ataupun terlalu panjang. Hal ini bisa untuk pedoman seorang dokter dalam pembuatan Gigi Tiruan ini³.

2. Diklat ILMU FAAL II oleh Dr Moeldjono dan Dr Bambang Budi Soesetyo dari UNAIR Surabaya.

3. Sharry JJ, Complete Denture Prosthodontics, The Blankiston Division McGraw Hill, 1986

2.2. PENGUKUR REGANGAN DAN PENGUKURAN PERUBAHAN-PERUBAHAN RESISTANSI YANG KECIL

2.2.1. Pendahuluan untuk Pengukuran Regangan

Pengukur regangan adalah kawat penghantar yang resistansinya berubah sedikit bila dipanjangkan atas atau dipendekkan. Perubahan panjang tersebut kecil, yaitu beberapa per juta dari satu centimeter. Pengukur regangan ditempatkan ke suatu struktur sedemikian rupa sehingga persen perubahan panjang dari pengukur regangan dan struktur itu sama.

Suatu pengukur jenis regangan terlihat dalam Gambar 2.2.1. Panjang aktif dari pengukur itu terletak disepanjang sumbu membujuranya. Pengukur regangan harus dipasang sedemikian rupa sehingga sumbu membujuranya terletak dalam arah yang sama dengan gerakan struktur yang harus diukur [lihat gambar 2.2.1(b) dan (c)]. Memanjangkan batang dengan tarikan akan memanjangkan penghantar pengukur regangan dan memperbesar resistansinya. Penekanan mengurangi resistansi pengukur karena panjang normal dari pengukur dikurangi.

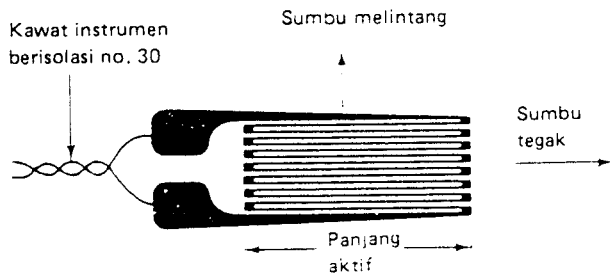
Tingkat sensitivitas regangan strain gage pada dasarnya dipengaruhi oleh dua faktor, yaitu:

- a> Perubahan dimensi kawat penghantar
- b> Perubahan tahanan spesifik pada specimen alloy kawat penghantar

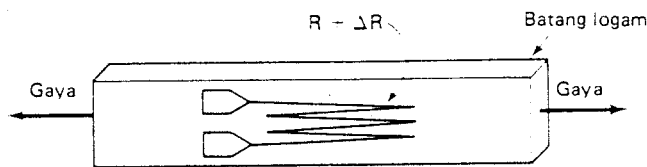
2.2.2. Menggunakan Data Pengukuran Regangan

Dalam pasal selanjutnya kita hanya akan mengukur *perubahan* resistansi pengukur δR saja. Faktor pengukur adalah

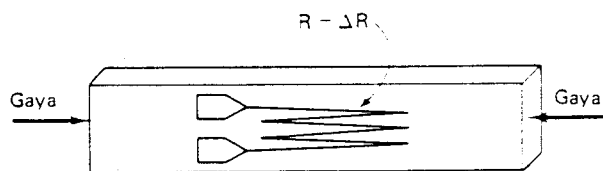
perbandingan dari persen perubahan resistansi sebuah pengukur terhadap persen perubahan panjangnya. Persen perubahan ini juga dinyatakan dalam desimal, jika perubahan $\delta R/R$ dibagi dengan faktor pengukur GR , hasilnya adalah *perbandingan perubahan panjang pengukur δL terhadap panjang mulanya L* .



(a) Pengukur regangan jenis lembaran logam



(b) Tarikan memanjangkan batang dan pengukur untuk memperbesar resistansi pengukur sebesar δR



(c) Penekanan pemendekan batang dan pengukur untuk mengurangi resistansi pengukur sebesar δR

Gambar 2.2.1.

Pengukur regangan untuk mengukur perubahan panjang struktur⁴

4. Robert F, Frederick F Driscoll, Penguat Operational dan Rangkaian Terpadu Linear, Penerbit Erlangga, 1985, hal 179

Tentu saja struktur dimana pengukur itu dipasang mempunyai $\delta L/L$ yang sama. Secara matematis dapat ditulis:

$$GR = \frac{\delta R/R}{\delta L/L} \quad (1)$$

GR biasa dikenal pula dengan sebutan faktor gage dari strain gage.

Perbandingan $\delta L/L$ mempunyai sebuah nama, yaitu disebut *regangan satuan*. Data regangan satuan ini yang dibutuhkan oleh para insinyur mesin. Mereka dapat menggunakan data regangan satuan ini bersama-sama dengan ciri-ciri bahan struktur (modulus kekenyalan) yang diketahui untuk mencari *tegangan* pada balok. *Tegangan* adalah besarnya gaya yang bekerja pada satu luas satuan. Satuan tegangan adalah Newton per meterpersegi (N/m).

2.2.3. Pemasangan Pengukur Regangan

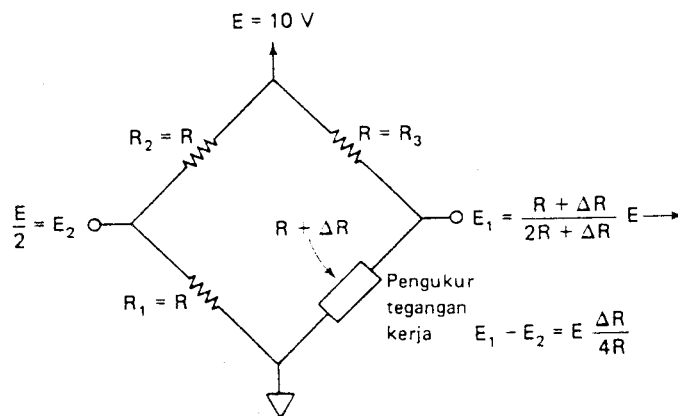
Sebelum memasang sebuah pengukur regangan permukaan balok yang dipasang harus bersih, halus dan dicuci alkohol, freon atau metil etil keton (MEK). Kemudian pengukur tersebut ditempelkan secara tetap ke permukaan yang dibersihkan dengan Eastan 910, lem epocy, perekat polymide, atau semen keramik. Prosedur pembuatan harus *betul-betul* dipatuhi.

Yang harus diukur adalah *perubahan* resistansi pengukur regangan δR dan perubahan ini kecil. δR mempunyai harga sebesar beberapa mili ohm. Teknik yang digunakan untuk mengukur perubahan resistansi kecil akan dibicarakan dalam pasal selanjutnya ini.

2.2.4. Pengukuran Perubahan Resistansi Kecil

2.2.4.1. Jembatan Resistansi Dasar

Pengukur regangan diletakkan pada salah satu lengan jembatan resistansi, seperti terlihat pada gambar 2.2.2. Misalkan bahwa pengukur regangannya tidak diregang, sedemikian rupa sehingga resistansinya = R . Misalkan juga bahwa R_1, R_2 , dan R_3 tepat sama dengan R .



Gambar 2.2.2.

Rangkaian Jembatan Wheatstone sederhana⁵

Dibawah syarat-syarat ini $E_1 = E_2 = E/2$ dan $E_1 - E_2 = 0$. Disebutkan bahwa jembatan itu menjadi *seimbang*. Jika pengukur regangannya ditekan, R akan bertambah sebesar δR dan perbedaan tegangannya akan diberikan oleh

$$E1 - E2 = E \frac{\delta R}{4R} \quad (2)$$

Persamaan (2) memperlihatkan bahwa E harus dibuat cukup besar untuk memaximumkan tegangan keluar diferensial jembatan, $E1 - E2$.

2.2.5. Efek-efek Panas pada Keseimbangan Jembatan

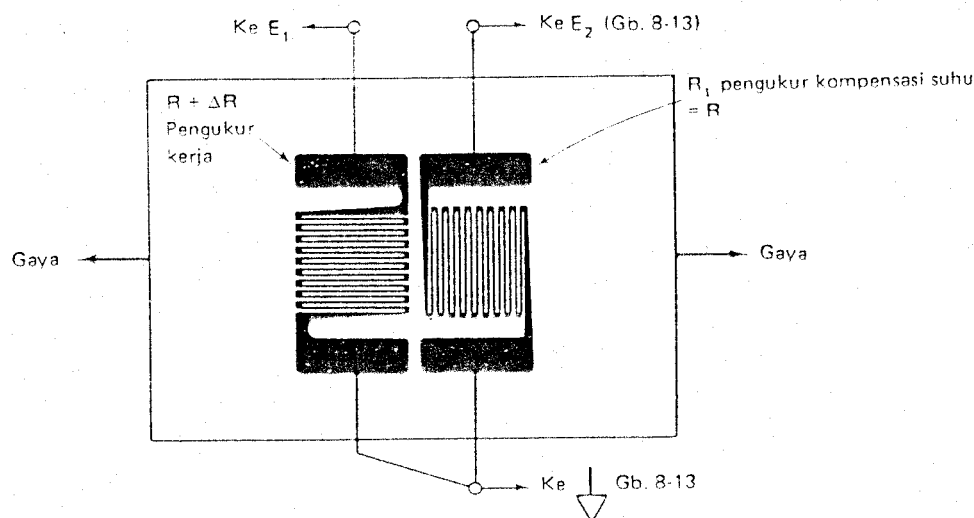
Bahkan jika kita berhasil menyeimbangkan rangkaian jembatan dari Gambar 2.2.2., rangkaian ini tidak akan menjadi tetap seimbang disebabkan perubahan suhu kecil dalam pengukur regangannya yang mengakibatkan perubahan resistansi yang sama dengan atau yang lebih besar dari yang disebabkan oleh regangan. Masalah ini dijawab dengan memasang pengukur regangan lainnya yang tepat di-sebelah pengukur regangan yang bekerja sedemikian rupa sehingga keduanya membagi lingkungan panas yang sama. Karena itu bersama berubahnya suhu, resistansi pengukur yang ditambahkan berubah tepat seperti resistansi pengukur yang bekerja pengukur kerja. Jadi pengukur regangan yang ditambahkan itu memberikan kompensasi suhu otomatis.

Pengukur kompensasi suhu dipasang dengan sumbu tegak yang *tegak lurus* terhadap sumbu tegak dari pengukur kerja, seperti terlihat dalam gambar 2.2.3.. Jenis susunan pengukur yang baku ini bisa didapat dari produsennya. Pengukur yang baru dihubungkan dengan menggantikan tempat tahanan $R1$ dalam rangkaian jembatan dari gambar 2.2.2.. Begitu jembatan ini telah diseimbangkan, δR dari pengukur kompensasi suhu dan pengukur kerja saling menjejaki yang lain untuk mempertahankan-

kan jembatan tersebut dalam keseimbangan. Setiap ketidakseimbangan jelas disebabkan oleh δR pengukur kerja akibat regangan.

2.2.6. Menyeimbangkan Sebuah Jembatan Pengukur Regangan

Untuk memeriksa cara kerja jaringan seimbang dalam gambar 2.2.4., misalkan bahwa tahanan-tahanan jembatan R_1 dan R_2 sama besar, sampai ketelitian meliputi kurang lebih 1 %. Resistansi pengukur regangan tersebut harus mempunyai resistansi yang sama meliputi beberapa miliohm jika pengukur yang bekerja tidak mengalami regangan.



Gambar 2.2.3.

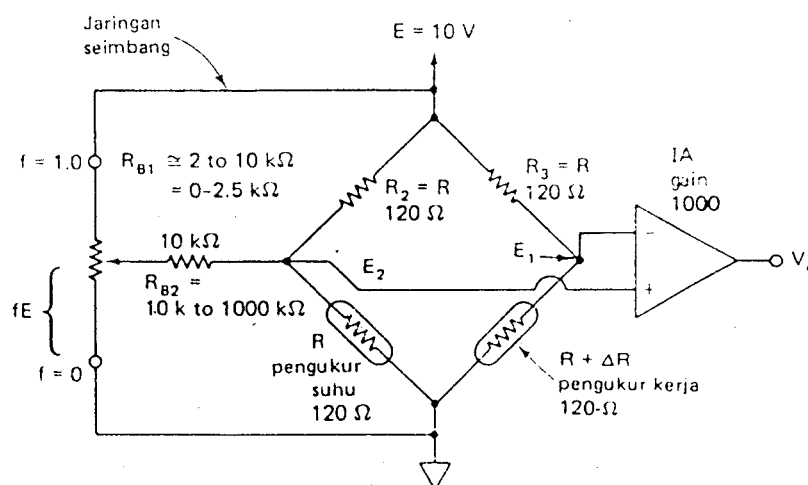
Pengukur kerja dan kompensasi suhunya⁶

Tahanan R_{b1} adalah pot linear 3/4 lilitan yang biasa. Resistansinya harus sekitar 1/10 atau lebih kecil dari taha-

6. Ibid, hal 192

nan R_{b2} sedemikian rupa sehingga tegangan fE hanya tergantung pada E dan berubah-ubah dari ujung yang satu ke ujung yang lainnya.

Tahanan R_{b2} harus dipilih lebih besar dari 10 kali masing-masing pengukur atau lebih dan juga 10 kali R_{b1} atau lebih. Dalam keadaan ini R_{b2} tidak membebani pekerjaan R_{b1} sebagai pembagi tegangan. Lagipula ukuran R_{b2} menentukan arus penyeimbangan maksimum yang dapat dialirkan ke dalam, atau diambil keluar dari simpul $E2$. f penyetel pot menentukan berapa banyak arus maksimum yang dialirkan kedalam atau diambil ke luar.

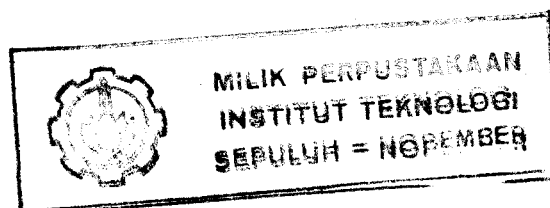


Gambar 2.2.4.

Teknik Jembatan Wheatsone yang lebih baik⁷

Kerja penyeimbangan diringkaskan dengan mengamati bahwa jika $f > 0.5$, suatu arus kecil dimasukkan kedalam simpul $E2$

7. Ibid, hal 184



dan mengalir melalui pengukur suhu ke ground, yang membuat $E2$ menjadi lebih positif. Jika $f < 0.5$ arus tersebut diambil dari simpul $E2$, yang memperbesar arus melalui $R2$, membuat $E2$ menjadi lebih positif.

Dalam susunan jembatan sesungguhnya, mulailah dengan $R_{b2} = 10 \text{ k}\Omega$ dan $R_{b1} = 1 \text{ k}\Omega$. Monitorlah V_O dari IA tersebut dan periksa tindakan penyeimbangannya. Jika perubahan V_O lebih besar dari yang anda inginkan, perbesar R_{b2} sampai $100 \text{ k}\Omega$ dan periksa kembali tindakan penyeimbangannya. Harga akhir dari R_{b2} dipilih berdasarkan percobaan dan tergantung pada derajat ketidakseimbangan antara $R2$ dan $R3$.

2.3. AMPLIFIER

2.3.1. Operational Amplifier (Op Amp) Ideal

Sebuah Op Amp dalam yang ideal mempunyai sifat-sifat sebagai berikut:

1. Penguatan loop terbuka (open loop gain) A_{ol} tak berhingga.
2. Impedansi Input Z_{in} tak berhingga, sehingga tidak membebani sinyal input.
3. Impedansi Output Z_{out} sama dengan nol, sehingga op amp ini dapat dibebani dengan beban impedansi berapapun.
4. Bandwidth yang tak berhingga, dapat menguatkan sinyal dari frekwensi rendah hingga frekwensi yang tinggi.
5. Differential input E_d sama dengan nol.
6. Karakteristik tidak berubah terhadap suhu.

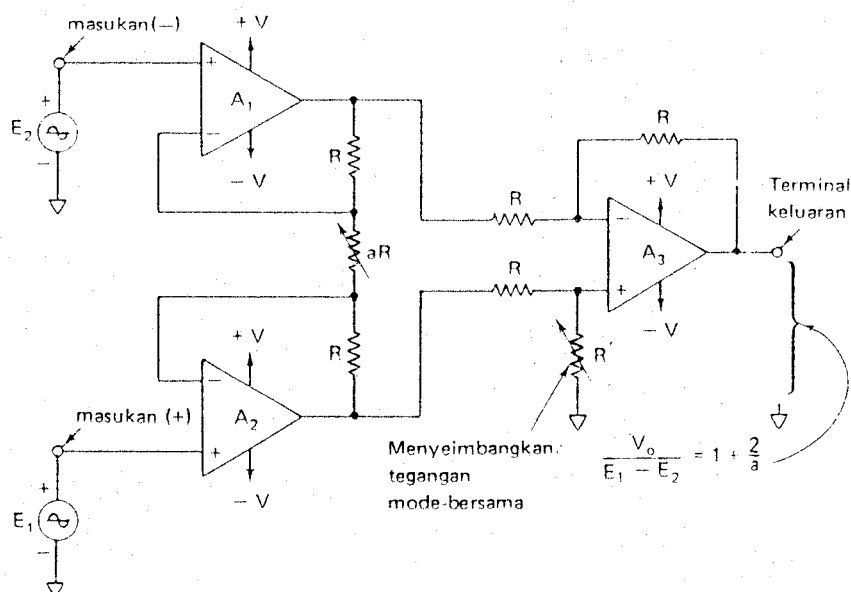
2.3.2. Instrumentasi Amplifier

Tegangan output suatu transducer pada umumnya sangat lemah sehingga pengaruh gangguan akan menjadi besar. Penguat inverting maupun non inverting biasanya kurang cocok karena penolakan terhadap ragam tunggal (CMRR) yang rendah, sehingga noise dan interferensi yang masuk pada kedua ujung inputnya tidak diredam. Sedangkan penggunaan penguat differensial dasar dapat mengatasi hal ini, tetapi masih ada kendala-kendala lain. Yaitu resistansi inputnya kecil dan penguatannya sulit diubah karena untuk melakukan hal ini perlu pengaturan nilai dua tahanan dan keduanya harus disesuaikan dengan cermat.

Kedua kendala ini dapat diatasi dengan menggunakan penguat instrumentasi. Rangkaian dasar instrumentasi amplifier dapat dilihat pada gambar 2.3.1.

Penguat ini dibangun dari 3 buah op-amp, yang pada dasarnya dibuat dengan menghubungkan sebuah penguat tersangga ke sebuah penguat differential dasar. Ciri-ciri penguat instrumentasi dapat diringkas sebagai berikut :

1. Penguatan tegangan dapat diatur dan cukup tinggi.
2. Resistansi inputnya sangat tinggi dan tidak berubah jika penguatannya diubah.
3. Mempunyai CMRR (Common Mode Rejection Ratio) atau penolakan terhadap ragam bersama yang besar.



Gambar 2.3.1.

Rangkaian Dasar Instrumentasi Amplifier⁸

2.4. ANALOG TO DIGITAL CONVERTER (ADC)

Pengubah analog ke digital adalah suatu rangkaian yang mengubah besaran analog ke besaran digital. Konversi ini untuk mempermudah pengolahan data dari sinyal yang akan dibaca.

Terdapat beberapa jenis ADC yang dapat dikelompokkan dalam empat kelompok seperti dibawah ini. Pemilihan ADC disesuaikan dengan penggunaan yang didalamnya diperhitungkan resolusi, waktu konversi dan ketelitian.

a> Successive Aproximation

ADC jenis successive aproximation atau yang berarti pendekatan berturut-turut, banyak digunakan terutama untuk interfacing dengan PC.

b> *Integrasi*

Pada ADC jenis ini terjadi konversi tidak langsung, yang pertama sebagai fungsi waktu, kemudian konversi dari fungsi waktu ke digital dengan menggunakan sebuah pencacah.

c> *Pencacah dan Servo*

Pada konverter jenis ini waktu konversinya tergantung dari perubahan tegangan input. Input analog dibandingkan dengan output DAC dengan input digital berasal dari pencacah.

d> *Paralel*

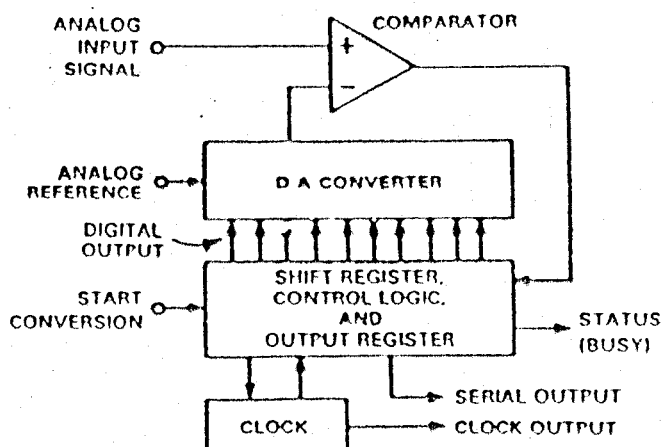
Konverter paralel atau flash menggunakan $2^n - 1$ pembanding. Penggunaan konversi paralel memungkinkan kecepatan hanya dibatasi waktu switching dari komparator gate. Bila input berubah, kode output berubah segera sehingga konverter jenis ini merupakan yang tercepat.

2.4.1. ADC Successive Aproximation

ADC Successive Aproximation mempunyai kelebihan-kelebihan antara lain mudah untuk interfacing dengan PC, waktu konversi tertentu, kecepatan konversi yang cukup tinggi dan memungkinkan untuk resolusi tinggi.

Cara kerja ADC ini adalah sbb. Konversi dilakukan dengan membandingkan input tak diketahui dengan sebuah tegangan atau arus presisi yang dibangkitkan oleh sebuah DAC, seperti pada gambar 2.4.1. Input DAC berasal dari output ADC. Pembandingan dilakukan bit demi bit dimulai dari MSB. Hal utama yang perlu diperhatikan pada ADC ini ialah input konversi

tidak boleh lebih dari 1 LSB selama konversi. Untuk mengatasi hal itu, pada sinyal-sinyal input yang berubah cepat digunakan *penguat sample and hold*. Penguat ini tidak diperlukan pada sinyal-sinyal yang berubah lambat.



Gambar 2.4.1.

Diagram Blok ADC Successive Aproximation⁹

2.5. MINIMUM SISTEM 8088

Mikroprosesor 8088 mempunyai arsitektur internal 18 bit dan mampu melaksanakan operasi-operasi pada operand 8 bit maupun 16 bit. Prosesor ini mempunyai 20 bit address bus, yang memberikan jangkauan pengaddressan memory sebesar 1 Mega byte dan pengaddressan peralatan input/output sebesar 64 Ki-lo byte.

Perancangan 8088 didasarkan pada kompatibilitas sistem,

⁹ Hall, Douglas V., Microprocessor and Digital System, McGraw Hill, 1983 hal 170

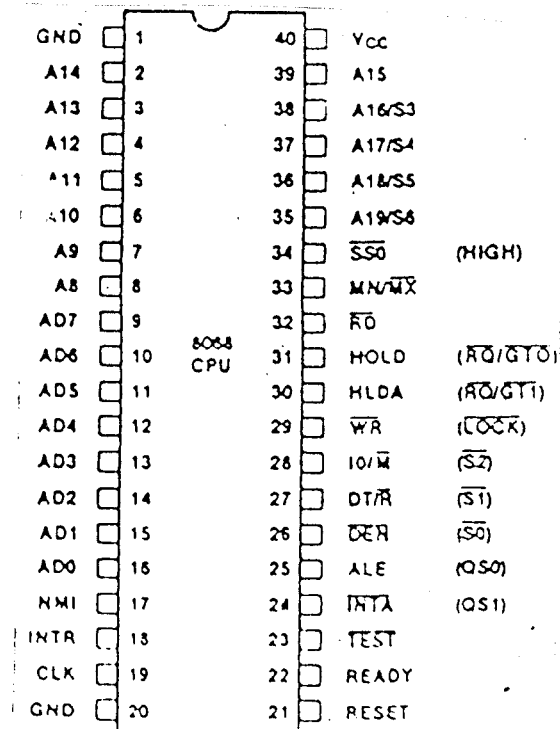
dan tersedia beraneka ragam co-processor untuk menunjang kemungkinan perancangan sistem terpadu, sejalan dengan hal ini, 8088 dirancang dengan data bus 8 bit, dalam upaya menunjang kompatibilitas dengan sistem-sistem yang deikenal sekitar prosesor 8080 terdahulu.

Mikroprosesor 8088 juga dirancang untuk memberikan kemudahan dalam perencanaan software. μP ini dapat melaksanakan operasi bit, byte, word, dan block dengan 24 mode operand addressing, operasi aritmetika 8 bit dan 16 bit dalam biner dan desimal. Pin-pin mikroprosesor 8088 dapat dilihat pada Gambar 2.5.1..

Berlainan dengan prosesor pendahulunya, μP 8088 mempunyai keunikan dalam hal pengaksesan memory. Prosesor ini menggunakan address bus 20 bit, yang memberikan jangkauan pengaddressan sebesar 1 Mbyte, tetapi register-register internal hanya mengandung maksimum 16 bit, yang memberikan jangkauan pengaddressan sebesar 64 Kbyte. Untuk menanggulangi keterbatasan ini memory dibagi dalam empat segment 64 Kbyte ; yaitu code segment yang berisi *instruction code* (kode perintah); data segment yang berisi data program; stack segment; dan extra segment, yang digunakan sebagai daerah data tambahan. Base address masing-masing segment tersimpan dalam pointer registers yang disebut segment register. Address yang sesungguhnya (absolute address) diperoleh dengan mengalikan segment register yang bersesuaian dengan 10 Mega dan ditambahkan dengan nilai *offset*.

Minimum sistem dengan 8088, seperti halnya dengan prosesor lain , terdiri dari tiga bagian utama, yaitu *Central*

Processing Unit (CPU), Memory Unit , dan Input/Output Unit. Bagian CPU terdiri dari μP 8088, *clock generator*, *buffer* dan *latch*, *wait state circuit*, dan *decoder* untuk control. Bagian memory terdiri dari *Random Access Memory (RAM)*, *Read Only Memory (ROM)*, dan *decoder* untuk menghasilkan sinyal *chip select* bagi RAM dan ROM. Bagian input/output (I/O) terdiri dari *I/O Interface* dan *decoder* yang menghasilkan sinyal *chip select* bagi I/O Interface.



Gambar 2.5.1.

Konfigurasi Pin μP 8088¹⁰

Minimum sistem dengan 8088 menggunakan 3-bus sistem architecture, yaitu address bus, dan control bus.

2.5.1. Internal Register dari μP 8088

Prosesor 8088 memiliki empat kelompok register 16 bit,

10. ———, Microprocessor and Peripheral Handbook Vol I, Intel Corporation, 1988 hal 2-89.

yaitu data register index register, pointer register, segment register, serta flag register dan instruction pointer.

Instruction pointer register menunjukkan lokasi memory dari instruksi yang akan dilaksanakan. Lokasi instruksi yang sebenarnya diperoleh dengan menggeser register CS 4 bit ke kiri, lalu ditambahkan isi register IP.

Flag register (2 byte) berisi status flags : carry flag, parity flag, auxiliary carry flag, zero flag, sign flag, over flow flag; dan control flag ; trap flag, interrupt flag, direction flag.

2.5.2. Basic Timing μ P 8088

Basic timing sangat diperlukan dalam perencanaan rangkaian buffer dan latch karena beberapa jalur pada 8088 dimultipleks. Selain itu basic timing juga diperlukan dalam perhitungan access time dari memory atau peralatan I/O. Basic Timing μ P 8088 untuk satu siklus pembacaan (read) dan penulisan (write) dapat dilihat pada gambar 2.2.2.

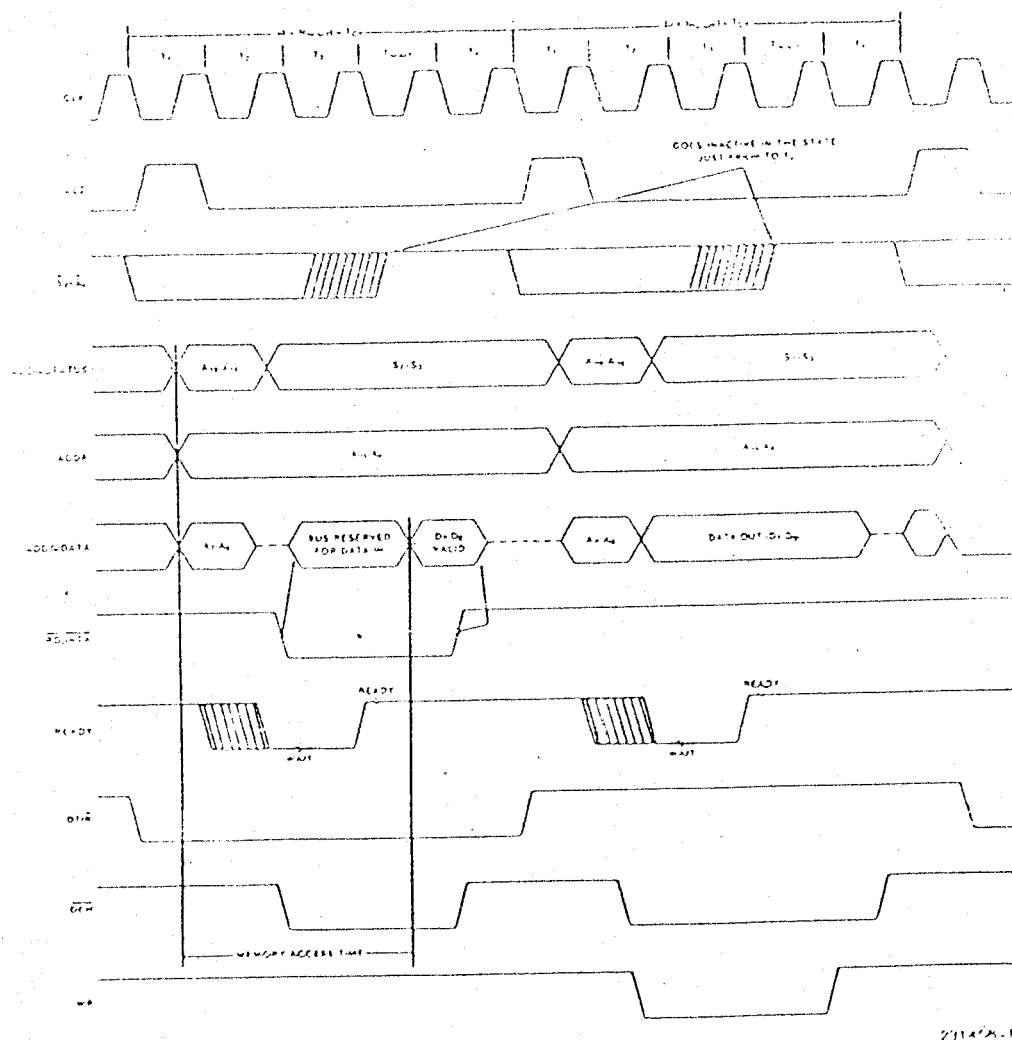
2.5.3 Read Only Memory (ROM)

Memory jenis ROM bersifat *non volatile*, artinya data yang disimpan tidak akan hilang ketika hubungan power supply ke ROM diputuskan. Ada beberapa jenis ROM tetapi yang paling banyak digunakan sekarang adalah jenis EPROM (Eraseable Programmable Read Only Memory).

2.5.4 Random Access Memory (RAM)

RAM digunakan untuk menyimpan data sementara waktu. Data pada RAM akan hilang bila supply tegangan terputus.

Berbeda dengan ROM, data dapat dikirim atau diambil dari RAM. RAM termasuk golongan volatile memory.



Gambar 2.2.2.

Basic Timing μP 8088¹¹

Ada dua jenis RAM, yaitu RAM statis dan RAM dinamis. RAM statis mempunyai keuntungan dalam hal kemudahan pengoperasiannya karena tidak diperlukan refresh Row Address Strobe (RAS), dan Column Address Strobe (CAS). Tetapi kerugiannya adalah harga dan ukuran perbit lebih besar daripada RAM dinamis. Bila hanya diperlukan memori berkapasitas kecil, penggunaan RAM statis lebih menguntungkan karena RAM dinamis memerlukan Dynamic RAM Controller. Pada rangkaian Hardware yang direncanakan digunakan RAM statis. Pada pembahasan selanjutnya, cukup disebut RAM saja.

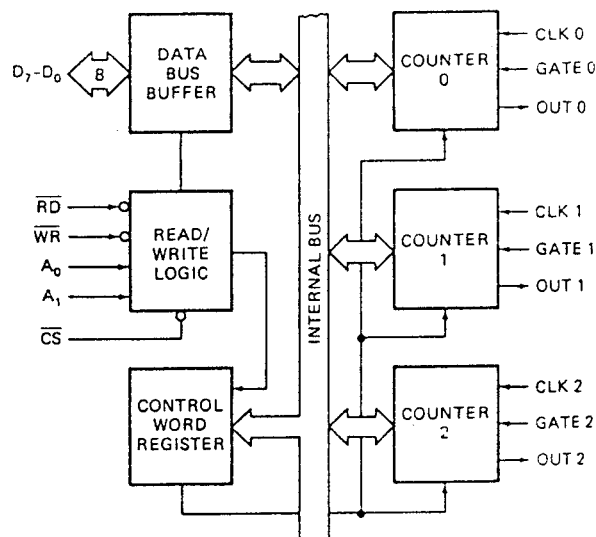
2.6. DECODER

Rangkaian decoder digunakan agar pada suatu saat hanya ada satu device yang bekerja, agar tidak terjadi konflik data. Yang dimaksud device disini dapat berupa RAM, ROM, atau Peripheral Interface. Rangkaian Decoder untuk RAM dan ROM berdasarkan pada memory mapping, sedang untuk peripheral interface berdasarkan I/O mapping.

Ada dua teknik yang biasa digunakan dalam decoding, yaitu teknik partial decoding dan full decoding. Pada teknik full decoding, seluruh zone bit terhubung ke rangkaian decoder. Dengan teknik ini suatu device hanya mempunyai satu lokasi memory atau dengan kata lain tidak ada duplikasi lokasi sehingga dapat menghemat lokasi memory. Pada teknik partial decoding tidak seluruh zone bit terhubung ke decoder yang mengakibatkan terdapat duplikasi lokasi untuk satu device.

2.7 PROGRAMMABLE INTERVAL TIMER 8253

Programmable Interval Timer (PIT) 8253 merupakan suatu device yang dapat digunakan untuk menghasilkan clock sesuai dengan yang diinginkan selama waktu tertentu. PIT ini dikontrol melalui program. Gambar 2.7.1. dibawah ini merupakan internal block PIT 8253.



Gambar 2.7.1.

Internal Block Diagram PIT 8253¹²

Data bus 8 bit pada PIT ini dihubungkan dengan data bus dari μP juga pin-pin kontrolnya seperti \overline{RD} dan \overline{WR} sedangkan chip select dihubungkan dengan decoder sesuai dengan

12. Hall, Douglas V., Microprocessors and Interfacing : Programming and Hardware, McGraw Hill, Singapore, 1986 hal 238

alamat yang diinginkan. PIT 8253 memiliki 4 address yang diatur me-lalui pin A0 dan A1, dimana masing-masing alamat di-tunjukkan sesuai tabel 2.1. dibawah.

Tabel 2.1.

Pemilihan Alamat pada PIT 8253

A0	A1	SELECT
0	0	COUNTER 0
0	1	COUNTER 1
1	0	COUNTER 2
1	1	CONTROL WORD REGISTER

PIT 8253 adalah counter 16 bit, bisa dalam bentuk BCD ataupun angka biner yang merupakan *down counter* atau penghitung mundur mulai dari angka tertinggi hingga 0h. Untuk mengawali pemrograman harus dikirimkan 8 bit control word pada alamat register control word. Control word gunanya untuk menentukan mode dan counter yang akan dipilih serta jenis kode yang dipilih dalam BCD atau biner. Format Control Word dapat dilihat pada gambar 2.7.2..

2.8. PROGRAMMABLE PERIPHERAL INTERFACE 8255

PPI 8255 merupakan periperal interface yang dapat di program fungsinya. PPI ini dibagi menjadi dua group, group A yang terdiri dari Port A (PA0 - PA7) dan Port C upper (PC4 - PC7); group B terdiri dari Port B (PB0 - PB7) dan Port C lower (PC0 -PC3). Internal Block diagram PPI 8255 dapat dilihat pada Gambar 2.8.1..

D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀
SC1	SC0	RW1	RW0	M2	M1	M0	BCD

SC – SELECT COUNTER:

SC1	SC0	
0	0	SELECT COUNTER 0
0	1	SELECT COUNTER 1
1	0	SELECT COUNTER 2
1	1	READ-BACK COMMAND (SEE READ OPERATIONS)

RW – READ/WRITE:

RW1	RW0	
0	0	COUNTER LATCH COMMAND (SEE READ OPERATIONS)
0	1	READ/WRITE LEAST SIGNIFICANT BYTE ONLY.
1	0	READ/WRITE MOST SIGNIFICANT BYTE ONLY.
1	1	READ/WRITE LEAST SIGNIFICANT BYTE FIRST, THEN MOST SIGNIFICANT BYTE.

M – MODE:

M2	M1	M0	
0	0	0	MODE 0 – INTERRUPT ON TERMINAL COUNT
0	0	1	MODE 1 – HARDWARE ONE-SHOT
X	1	0	MODE 2 – PULSE GENERATOR
X	1	1	MODE 3 – SQUARE WAVE GENERATOR
1	0	0	MODE 4 – SOFTWARE TRIGGERED STROBE
1	0	1	MODE 5 – HARDWARE TRIGGERED STROBE

BCD:

0	BINARY COUNTER 16-BITS
1	BINARY CODED DECIMAL (BCD) COUNTER (4 DECADES)

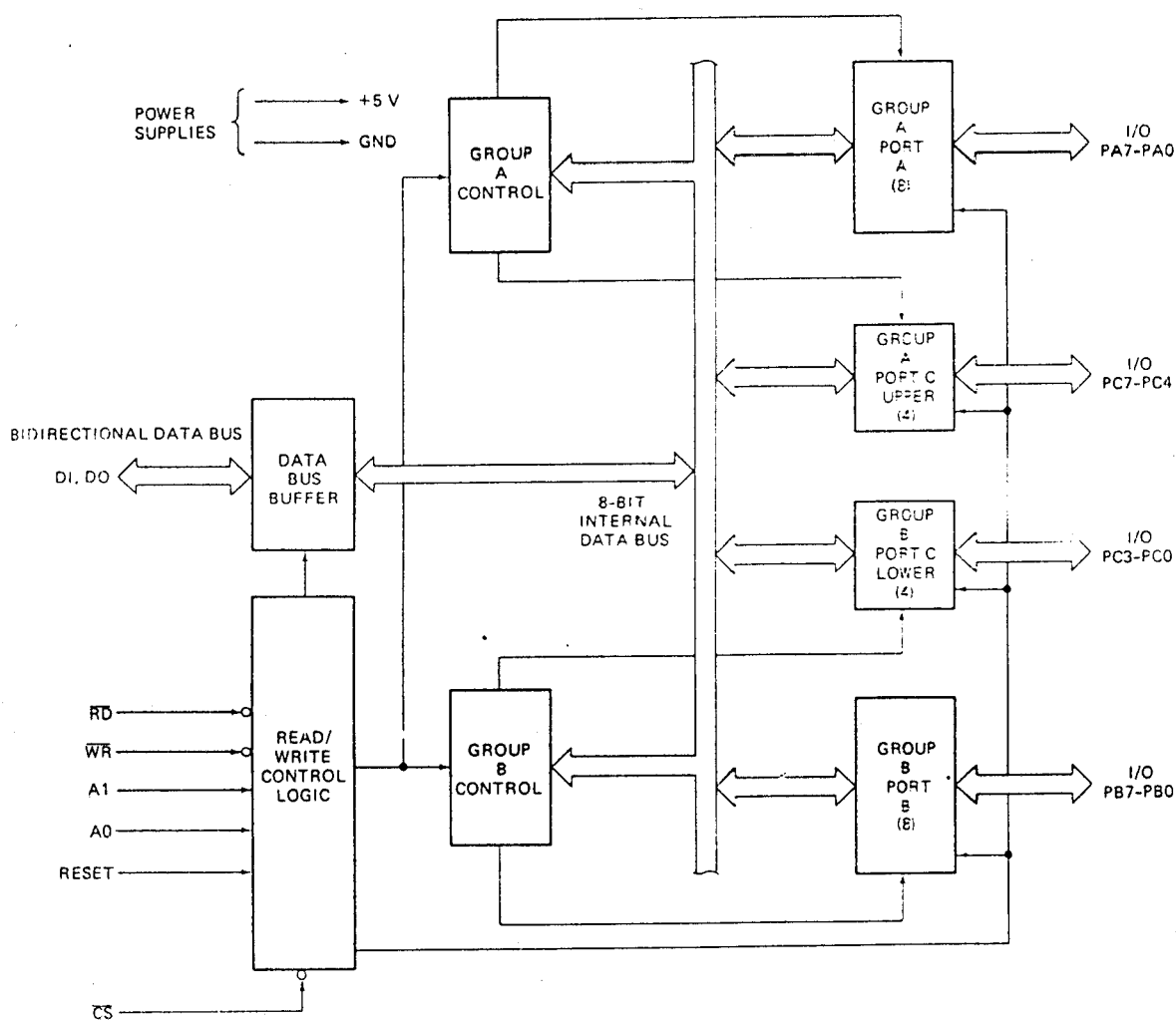
NOTE: DON'T CARE BITS (X) SHOULD BE 0 TO INSURE COMPATIBILITY WITH FUTURE INTEL PRODUCTS.

Gambar 2.7.2.

Format Control Word PIT 8253¹³

PPI 8255 mempunyai empat buah register, register mana yang bekerja ditentukan oleh kombinasi A0 dan A1. Lokasi dan fungsi masing-masing register pada tabel 2.1. Control Word Register berfungsi menentukan fungsi dari setiap port dan me-

entukan mode yang digunakan. PPI 8255 mempunyai tiga mode yang dapat dioperasikan, yaitu mode 0 ; basic input/output, mode 1 ; strobed input/output, dan mode 2 ; bidirectional bus. Format dari control word dapat dilihat pada gambar 2.8.2.



Gambar 2.8.1

Internal Blok Diagram PPI 8255¹⁴

Tabel 2.2.

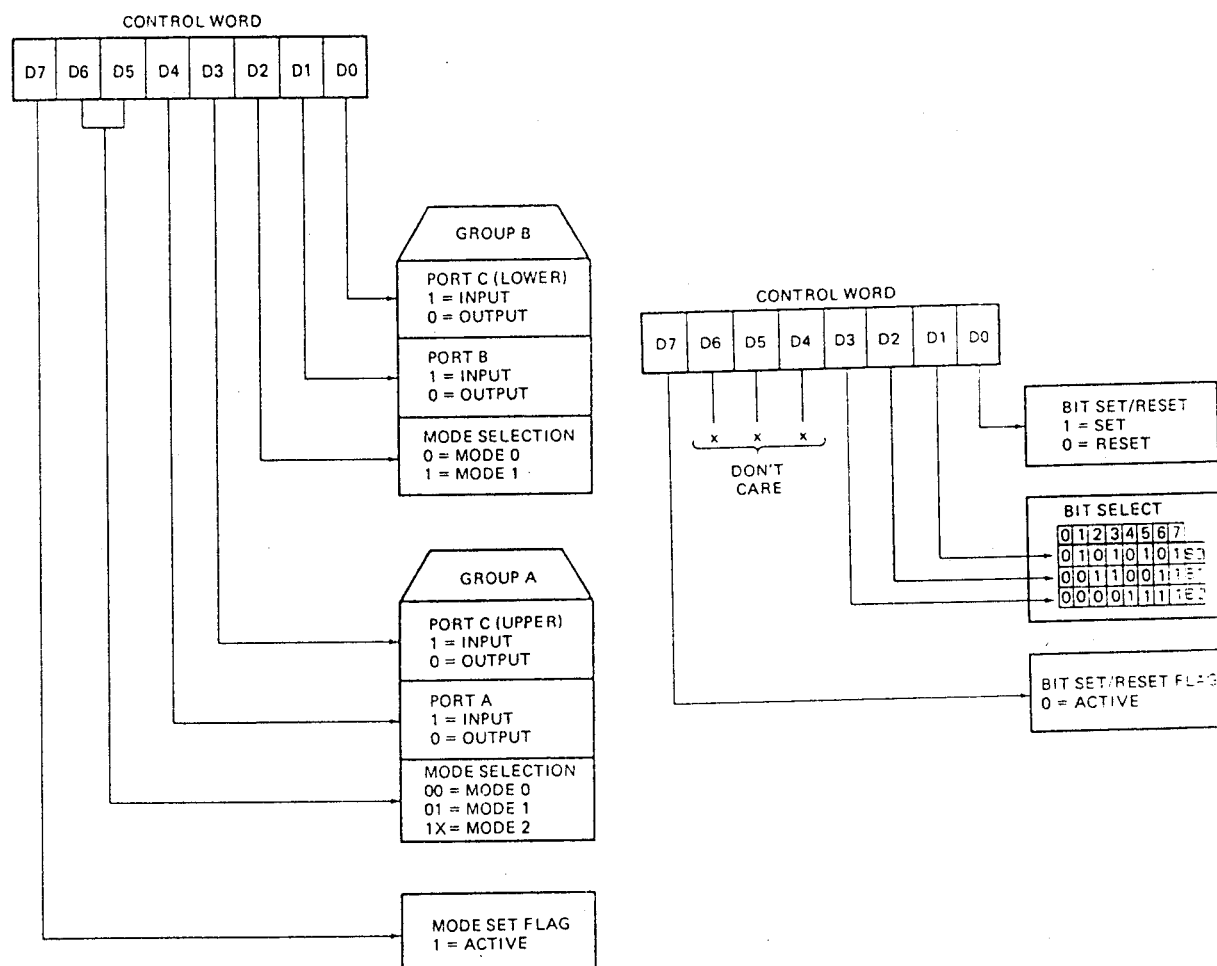
Lokasi dan Fungsi register pada 8255

A ₁	A ₂	RD ₋	WR ₋	CS ₋	INPUT OPERATION (READ)
0	0	0	1	0	PORT A -> DATA BUS
0	1	0	1	0	PORT B -> DATA BUS
1	0	0	1	0	PORT C -> DATA BUS
					OUTPUT OPERATION (WRITE)
0	0	1	0	0	DATA BUS -> PORT A
0	1	1	0	0	DATA BUS -> PORT B
1	0	1	0	0	DATA BUS -> PORT C
1	1	1	0	0	DATA BUS -> CONTROL
					DISABLE FUNCTION
X	X	X	X	1	DATA BUS -> TRI STATE
1	1	0	1	0	ILLEGAL CONDITION
X	X	1	1	0	DATA BUS -> TRISTATE

2.9 PRIORITY INTERRUPT CONTROLLER (PIC) 8259

Priority Interrupt Controller PIC 8259 adalah suatu device yang dihubungkan dengan μP dan dikontrol melalui program untuk melakukan interupsi pada pelaksanaan suatu program. Internal Bloknnya dapat dilihat pada gambar 2.9.1.

Data bus pada PIC terhubung dengan data bus pada μP . Data bus mengijinkan μP mengirim control word ke 8259 dan membaca status. RD₋ dan WR₋ mengontrol transfer ini saat CS₋ "low". Data bus juga mengijinkan 8259 mengirimkan type interrupt ke μP . Pengalamatan PIC ini dipilih dari output decoder yang masuk ke pin CS₋, sebuah PIC memiliki dua buah alamat yang diatur dari pin A0.

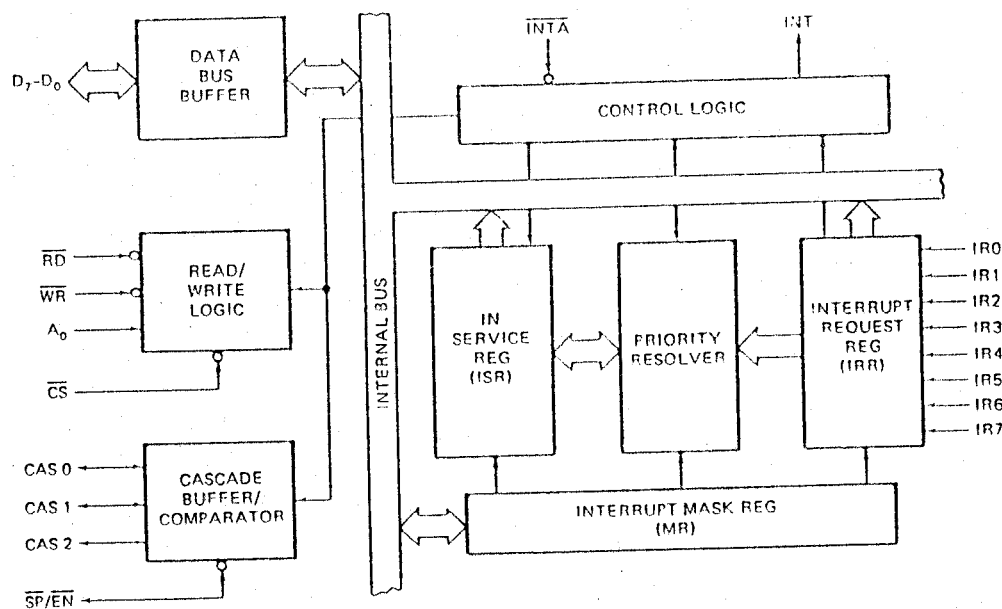


Gambar 2.8.2.

Format Control Word PPI 8255¹⁵

PIC 8259 memiliki 8 buah pin interrupt dimana prioritasnya interruptnya dapat diatur menurut mode yang dipilih. Antara lain ke-delapan interrupt memiliki tingkat prioritas

yang sama atau yang lebih kecil tingkat prioritasnya lebih tinggi, dengan kata lain IRQ0 memiliki tingkat prioritas tertinggi sedangkan IRQ7 memiliki tingkat prioritas terendah dan sebagainya. Ini berarti apabila CPU sedang menjalankan interrupt IRQ2, interrupt yang lebih besar seperti IRQ3 atau lebih tidak dapat menginterrupt tetapi IRQ1 dan IRQ0 dapat menginterrupt interupsi dari IRQ2 tersebut. PIC juga menyediakan *cascade lines* CAS0, CAS1 dan CAS2.

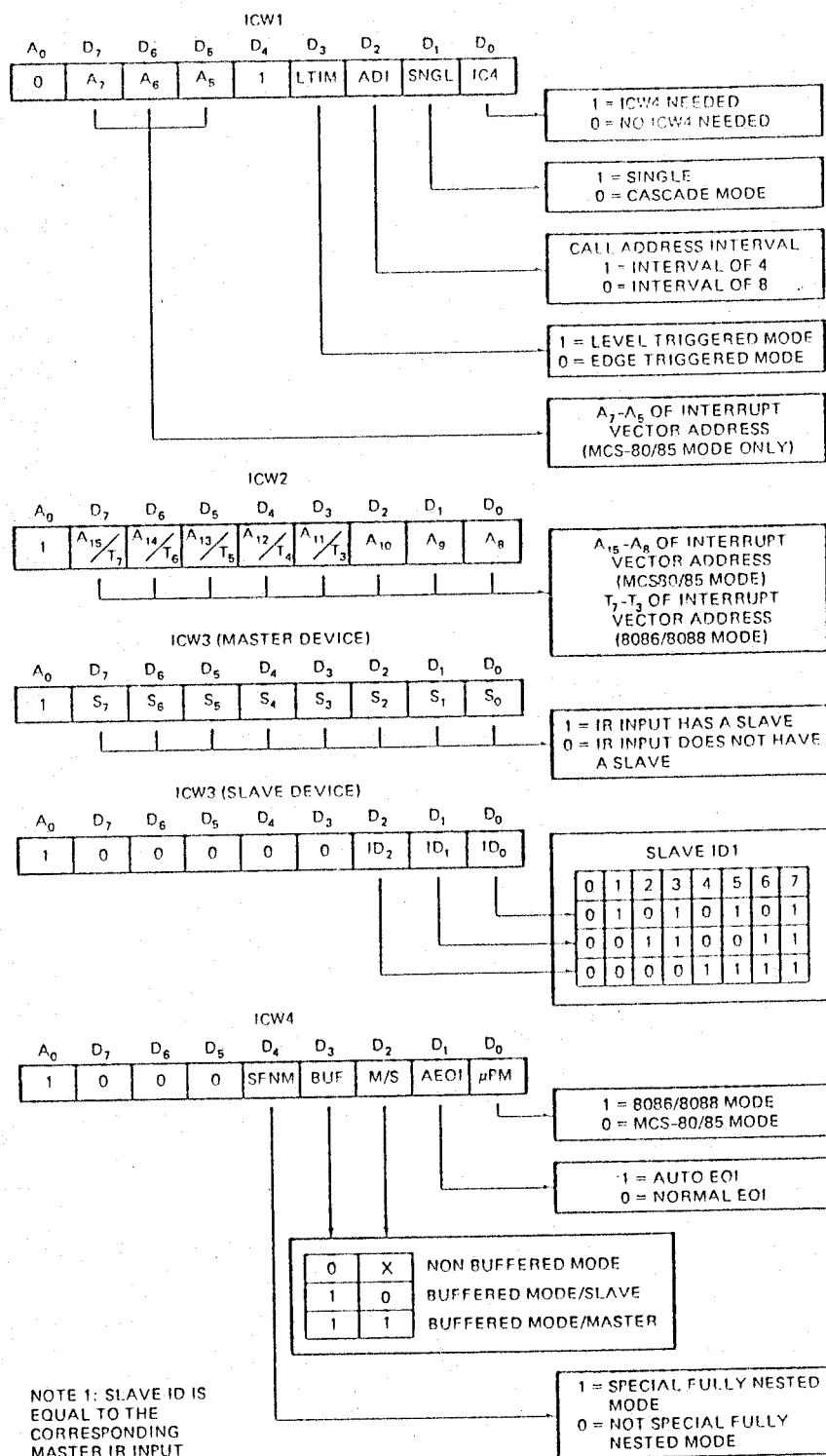


Gambar 2.9.1.

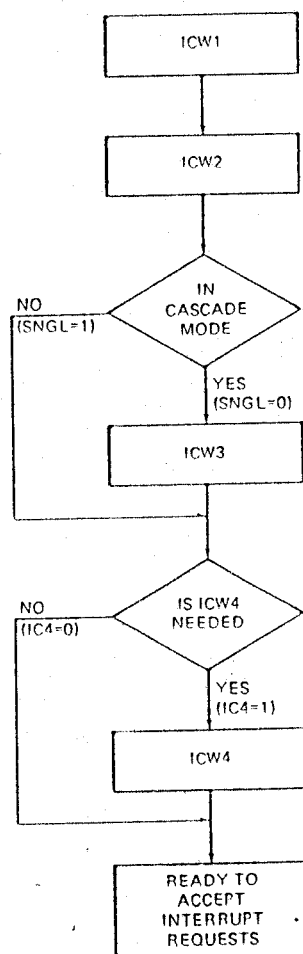
Internal Block Diagram PIC 8259¹⁶

16. Ibid hal 250

PIC 8259 ini memiliki 2 jenis Control Word yaitu *Initialization Command Word* (ICW) dan *Operational Command Word* (OCW). Format ICW dapat dilihat pada Gambar 2.9.2 sedang format OCW pada Gambar 2.9.3..



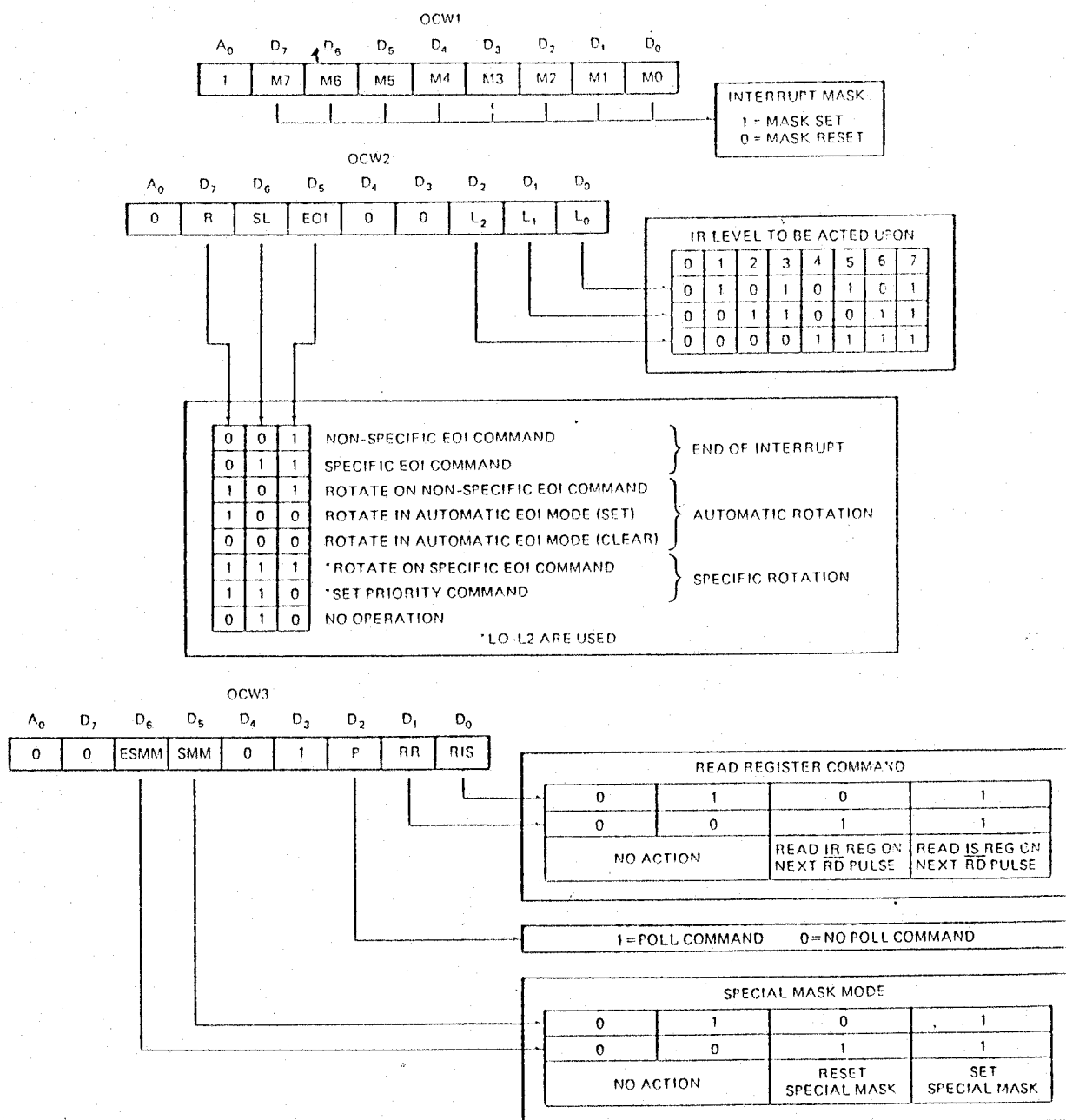
(a)



(b)

Gambar 2.9.2.

Format Initialization Command Word



Gambar 2.9.3.

Format Operational Command Word

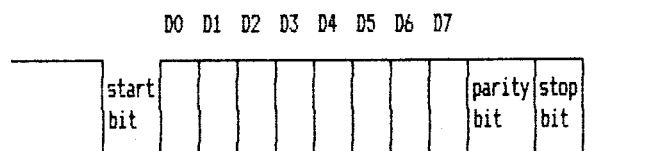
2.10. PRINSIP KOMUNIKASI DATA SERIAL

Pada komunikasi serial, data ditransmisikan melalui satu kabel, satu bit pada suatu waktu. Jadi bit-bit yang ada pada suatu byte data harus menunggu gilirannya untuk dipindahkan. Ada dua metode komunikasi seri, yaitu sinkron dan asinkron.

Pada komunikasi sinkron data dikirim dalam suatu blok-blok dengan kecepatan tetap. Kecepatan yang tetap ini diperoleh dengan mensinkronkan pengirim dan penerima. Tidak ada bit-bit khusus, sebab penerima mengetahui bahwa setiap 8 bit yang diterima setelah sinkronisasi, adalah mewakili satu byte data.

Pada komunikasi asinkron, setiap data karakter mempunyai satu bit khusus yang menandai awal data dan satu atau dua bit lain yang menandai akhir data. Karena setiap data karakter mempunyai tanda sendiri, maka data dapat dikirim pada sembarang waktu. Gambar 2.10.1 memperlihatkan bagaimana suatu data karakter dikirim. Ketika tidak ada yang dikirimkan, level sinyal adalah tinggi. Keadaan ini disebut marking state. Awal dari data ditandai dengan keadaan sinyal menjadi rendah selama satu bit, bit ini dinamakan bit start. Kemudian bit demi bit data dikirim mulai dari *least significant bit* (*LSB*). Jumlah bit yang terkandung dalam satu karakter data tergantung dari sistem yang digunakan. Bit paritas digunakan untuk mendeteksi kesalahan data yang diterima. Pada akhir data karakter, sinyal menjadi tinggi selama satu atau dua bit, bit ini dinamakan bit stop. Bit stop ini mengemba-

likan sinyal pada kondisi marking state dan siap memulai data baru.



Gambar 2.10.1.

Format Data Karakter pada Komunikasi Asinkron

Misalkan dipakai konfigurasi 1 bit start, 7 bit data, dan dua bit stop, tanpa bit paritas, berarti 10 bit dikirim untuk 1 byte. Jadi jika kecepatan transmisi adalah 9600 bit per detik, maka akan dipindahkan 960 byte/detik.

2.10.1 Standart Komunikasi RS-232

Komunikasi data melalui port komunikasi pada IBM PC menggunakan standard protokol RS_232. Standard ini dikembangkan oleh *Electronic Industries Association*. Sistem ini semula digunakan untuk mengatur hubungan antara komputer dan modern. Tetapi perkembangan selanjutnya, RS-232 juga digunakan untuk piranti pheripheral lainnya misal printer, plotter, ataupun mouse.

RS-232 menggunakan logika negatif, yaitu tingkat logika 1 berada dalam daerah tegangan -3 Volt sampai -15 Volt, dan tingkat logika 0 berada dalam daerah 3 Volt sampai 15 Volt.

2.10.2 Hubungan RS-232

RS-232 mempunyai 25 pin, namun tidak mutlak harus digunakan semua, tergantung dari sistem yang digunakan. Dari ke-25 pin tersebut dapat digolongkan atas 4 fungsi yaitu : ground, pertukaran data, kontrol, dan pewaktu.

Tabel 2.3. memperlihatkan nama sinyal dari DTE (*Data Terminal Equipment*). Pin yang paling umum digunakan dalam suatu sistem yang sederhana adalah sinyal TxD, RxD, dan sinyal handsaking. Tetapi pada beberapa sistem tersebut tidak diperlukan sinyal handsaking. Pin sinyal TxD (no.2) digunakan untuk transmisi data serial, sedang pin sinyal RxD (no.3) digunakan untuk menerima data serial. Sinyal-sinyal *handsaking* umumnya digunakan antara komputer dan modem.

Tabel 2.3.

Pin-pin pada RS-232

Nama	Deskripsi	no pin	arah
TxD	Transmitted data	2	out
RxD	Received data	3	in
RTS	Request to Send	4	out
CTS	Clear to Send	5	in
DSR	Data Set Ready	6	in
GND	Signal Ground	7	-
CD	Carrier Detect	8	in
DTR	Data Terminal Ready	20	out

2.10.3 Asynchronous Communications Ports Pada IBM PC

Pada komputer IBM PC terdapat suatu interface serial komunikasi. Port komunikasi ini menggunakan standard RS-232

yang merupakan fasilitas standard bagi tiap komputer IBM PC. Interface ini menempati alamat yang telah ditentukan oleh sistem IBM PC (Tabel 2.4.).

Tabel 2.4.

Alamat Interface serial komunikasi

Designation	Range Alamat
COM1:	3F8H - 3FFH
COM2:	2F8H - 2FFH
COM3:	3E8H - 3EFH
COM4:	2E8H - 2EFH

Port 3F8h, digunakan untuk mengirim dan menerima data, dan dapat digunakan juga (Jika bit 7 port 3FBh di set) untuk menentukan baud rate peralatan komunikasi. Bilangan pembagi tersebut digunakan untuk membagi sistem clock dari peralatan komunikasi untuk mendapatkan baud rate yang diinginkan.

Port 3F9h juga digunakan untuk menentukan byte tinggi (Jika bit 7 port 3FBh di set) bagi pembagi baud-rate. Jika bit 7 port 3F9h di-clear maka akan berfungsi sebagai interrupt enable register.

Port 3FBh, line control register, digunakan untuk menentukan format data yang dikirim atau yang diterima melalui port komunikasi. Port 3FDh, line status register, digunakan untuk menandai keadaan transfer data.

Sebelum mengoperasikan port komunikasi, maka harus dilakukan inisialisasi terlebih dahulu. Inisialisasi ini digu-

nakan untuk menentukan Baud rate sistem, format data, dan metode komunikasi dengan PC. setelah dilakukan inisialisasi maka port komunikasi digunakan untuk mengirim maupun menerima data.

Tabel 2.5.

Line Control Register - Port 3FBh (atau 2FBh, 3EBh, 2EBh)

Bits 76543210	Arti
0	Akses normal ke port 3F8h/3F9h
1	Memakai port 3F8h/3F9h utk baud-rate divisor
1	Kondisi Transmit Break
0	Parity pada bit 4
1	Parity beroperasi normal
0	Parity ganjil
1	Parity genap
0	Parity disabled
1	Parity enabled
0	1 stop bit
1	2 stop bit
00	5-bit data
01	6-bit data
10	7-bit data
11	8-bit data

Tabel 2.6.

Line Status Register - Port 3FDh (2FDh, 3Edh, atau 2Edh)

Bits 76543210	Arti
0	Akses normal ke port 3F8h/3F9h
0	Transmitter Shift Register penuh
1	Transmitter Shift Register kosong
0	Transmitter Holding Register penuh
1	Transmitter Holding Register penuh
1	Kondisi Break terdeteksi
1	Framming Error
1	Parity Error
1	Overrun error
0	Tidak ada karakter siap
1	Data siap diterima

2.10.4 Inisialisasi Port Komunikasi

Dalam inisialisasi, data-data yang digunakan untuk inisialisasi dikirim pada line control register. Misalkan pada suatu sistem komunikasi digunakan baud rate 300, 600, 1200, atau 2400 bps. Sedangkan format data yang dikirimkan adalah 7 bit data, parity genap, 1 start bit dan 2 stop bit, dan mode komunikasi dengan PC ialah pooling.

2.10.5 Membaca Data Dari Port Komunikasi

Setelah dilakukan inisialisasi maka port komunikasi telah siap menerima atau mengirim data. Untuk dapat menerima data maka harus dilakukan test untuk kesiapan data dan adanya kesalahan misalnya parity atau overrun. Test ini dilakukan pada Line status register. Jika semua test memenuhi syarat maka data yang valid dapat dibaca dari data register.

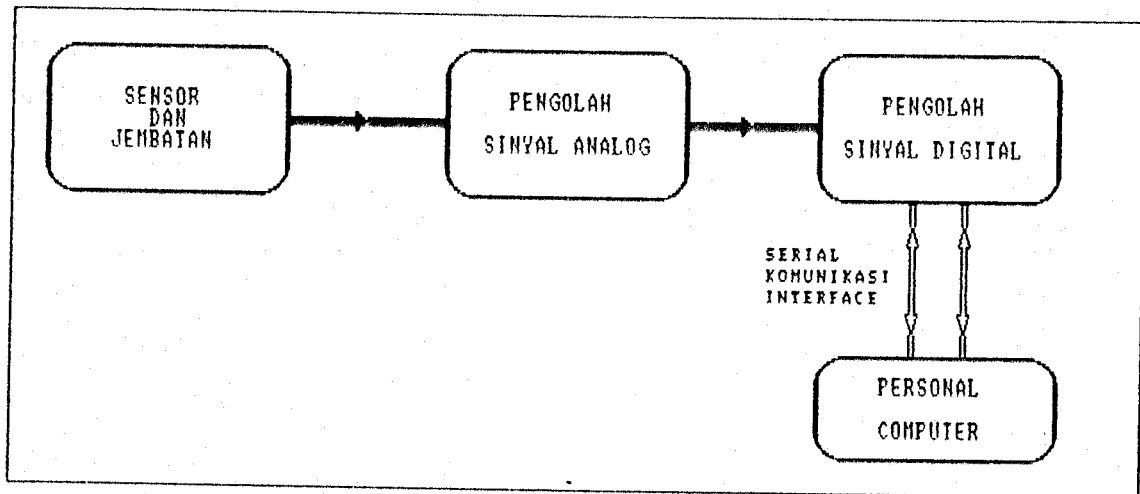
2.10.6 Mengirim Data Melalui Port Komunikasi

Untuk dapat mengirim data maka harus dilakukan test untuk kesiapan register data pada port komunikasi agar tidak terjadi kesalahan pengiriman data. Status register yang dilihat ialah kosongnya transmit ialah kosongnya transmit shift register dan transmit holding register. Test ini dilakukan pada Line status register. Jika semua test memenuhi syarat maka data dapat dikirim ke data register.

BAB III

PERENCANAAN HARDWARE DAN SOFTWARE

Urutan langkah pengerjaan hardware secara umum dapat dilihat dari blok diagram dibawah ini.



Gambar 3.1

Blok Diagram Perencanaan Hardware

Dengan diberikannya tekanan pada ujung plat, output tegangan dari Jembatan Wheatsone juga berubah. Output tegangan ini kemudian diproses pada bagian pengolah sinyal analog yang kemudian dikonversi menjadi sinyal digital dan hasilnya ditampilkan pada display.

3.1 SENSOR PADA ALAT UKUR

3.1.1. PLAT DAN STRAIN GAUGE

Sensor atau tranducer yang direncanakan untuk dipakai pada alat ukur ini adalah strain gauge yang dilekatkan pada plat logam. Sedangkan sistem yang digunakan adalah sistem *bending*, yaitu salah satu ujung plat diklem kemudian sisi

yang lain dikenai tekanan.

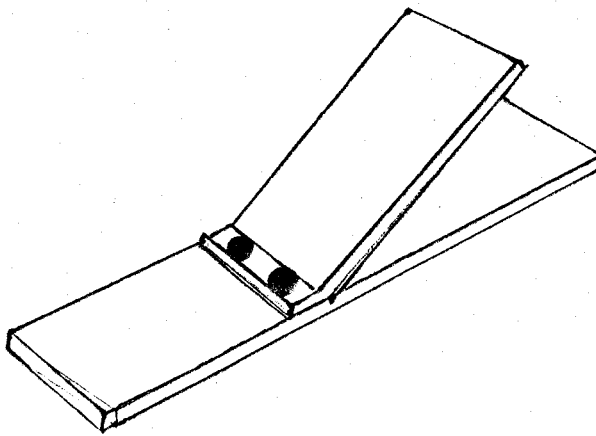
Plat logam yang akan digunakan disini adalah Stainless steel setebal 2 mm. Plat ini diklem/dilekatkan pada plat sejenis dengan ketebalan 4 mm sebagai alas. Plat yang tipis diatur sehingga membentuk sudut terhadap plat yang lebih tebal. Pada ujung plat yang tipis inilah akan diberikan tekanan.

Secara ideal plat yang diinginkan adalah plat logam yang lentur (tetapi jika diberikan tekanan sampai sebesar 50 kg plat masih dapat kembali seperti semula) sebagai tempat melekatnya strain gauge dan plat yang keras (tidak mengalami perubahan akibat adanya tekanan) sebagai alas. Kedua plat memiliki ketebalan setipis mungkin sehingga mempermudah pasien.

Disini direncanakan 2 macam plat , yaitu 1 plat biasa berbentuk huruf I untuk sensor pada gigi seri dan 1 plat berbentuk huruf Y sebagai sensor pada kedua sisi gigi geraham. Pada pemasangannya dilekatkan karet pelapis pada alas dan ujung tekanan (bagian yang digigit). Hal ini dilakukan agar gigi pasien tidak langsung menekan logam. Karet yang dipilih adalah karet yang cukup keras sehingga tidak terlalu meredam tekanan pada plat.

Saat diberikan tekanan pada ujung sensor plat stainless stell, ujung plat tersebut akan bergerak turun sesuai arah tekanan sedangkan ujung lain tetap karena telah diklem pada alas plat. Gerakan ini menyebabkan perubahan panjang pada plat dan dideteksi dengan menggunakan transducer strain gauge yang ditempelkan pada bagian paling kritis pada plat. Peru-

bahan panjang dari strain gauge akan menyebabkan resistansi strain gauge yang terangkai pada Jembatan Wheatstone berubah.



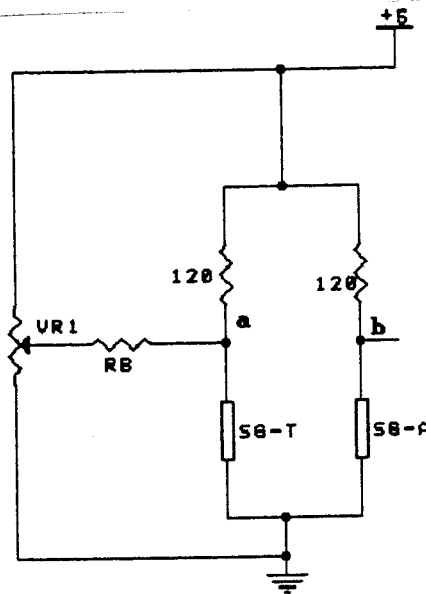
Gambar 3.1.1

Bentuk Alat Ukur

3.1.2 RANGKAIAN JEMBATAN WHEATSTONE

Sebagai perubah sensor tahanan menjadi tegangan digunakan rangkaian jembatan Wheatstone. Pada jembatan Wheatstone terdapat empat buah lengan tahanan, apabila keempat lengan jembatan dipasang strain gauge, sensitifitas dan ketelitian alat ukur akan semakin baik dibandingkan jika hanya dipasang pada dua lengan saja. Karena keterbatasan tempat pada plat, pada Tugas Akhir ini untuk setiap jembatan hanya dipasang 2

buah strain gauge. 1 buah sebagai transducer atau sensor penghasil tegangan yang dipasang searah panjang plat, sedangkan sebuah yang lain sebagai kompensasi suhu yang dipasang melintang panjang plat. Bentuk rangkaian jembatan dapat dilihat pada Gambar 3.1.2.



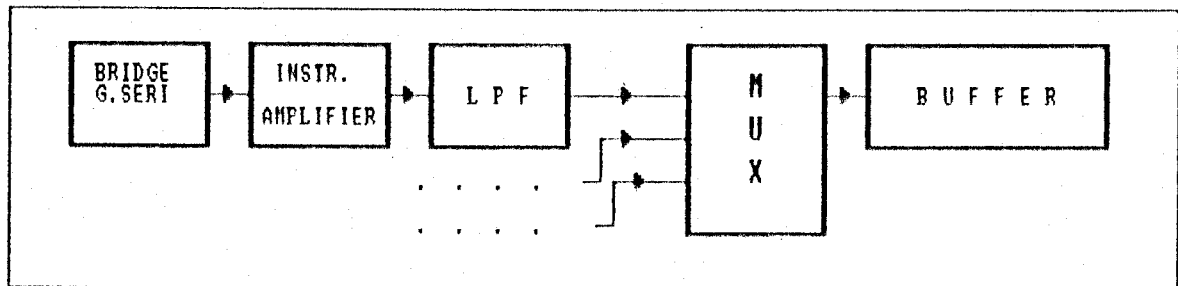
Gambar 3.1.2.

Rangkaian Jembatan Wheatstone

Resistor variabel VR1 digunakan sebagai penyeimbang jembatan (untuk mengatur tegangan jepit $V_{ab} = 0$ V). Dengan memberikan tekanan pada ujung plat akan menyebabkan perubahan panjang plat demikian panjang strain gauge juga akan berubah yang menyebabkan resistansi strain gauge ini berubah. Dari rangkaian jembatan diatas hanya dapat dilihat bahwa kenaikan resistansi strain gauge menyebabkan tegangan jepit V_{ab} menjadi naik. Perubahan tegangan jepit V_{ab} ini sangat kecil (dalam orde mili Volt).

3.2 PENGOLAH SINYAL ANALOG

Urut-urutan jalannya sinyal pada pengolah sinyal analog ditunjukkan pada Gambar 3.2.1 dibawah ini.



Gambar 3.2.1.

Blok Diagram Pengolah Sinyal Analog

3.2.1 INSTRUMENTASI AMPLIFIER

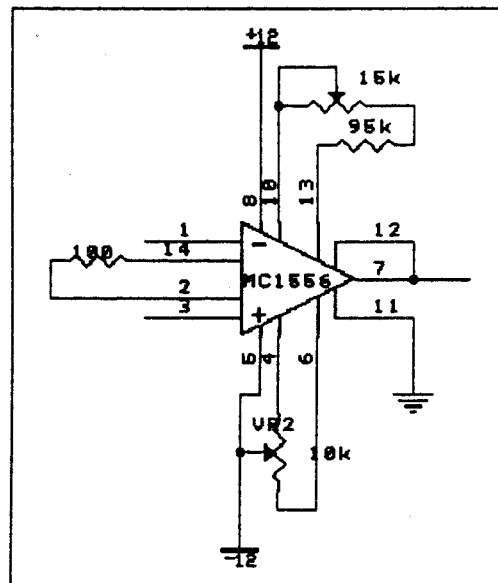
Output dari sensor, dalam hal ini jembatan wheatstone yaitu tegangan jepit V_{ab} adalah sangat kecil. Sehingga untuk mendapatkan sinyal yang cukup besar untuk dapat dikonversi oleh ADC, sinyal ini diperkuat dengan menggunakan suatu rangkaian Amplifier, yaitu rangkaian Instrumentasi Amplifier. Rangkaian Instrumentasi Amplifier ini menggunakan Precision Instrumentasi Amplifier AD521JD.

Penguatan (Gain) dari Instrumentasi Amplifier AD521 dapat diatur mulai dari 0 - 1000. Besarnya dapat dihitung dengan rumus :

$$\frac{V_{out}}{V_{in}} = \frac{R_s}{R_g}$$

Gain dapat diatur dengan merubah-ubah nilai R_g . Dibawah ini tabel besarnya gain dan nilai R_g .

Gain	R_g
0,1	1 M Ω
1	100 k Ω
10	10 k Ω
100	1 k Ω
1000	100 Ω



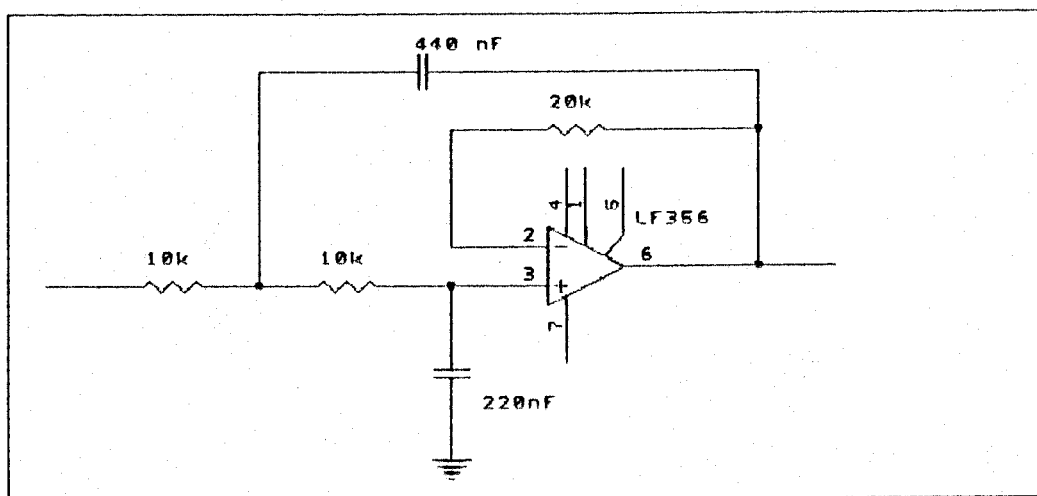
Gambar 3.2.2

Operating Connections AD521

Diperkirakan output dari transducer maksimum 7 mV, sedangkan kemampuan penerimaan input pada ADC 0804 direncanakan 5 Volt. Untuk itu penguatan pada Instrumentasi Amplifier dibuat 700 kali. Hal ini berarti dipilih besarnya $R_g = 100 \Omega$ dan mengatur R_s untuk mendapatkan penguatan 500 kali ini.

3.2.2 FILTER

Sebagai penyaring noise frekwensi tinggi digunakan sebuah Low Pass Filter (LPF). LPF yang direncanakan adalah LPF orde 2 (40 db) dengan frekwensi cut off 50 Hz. Untuk mendapat hasil yang baik dipilih amplifier LF356. Gambar rangkaian tampak seperti Gambar 2.3.



Gambar 3.2.3

Low Pass Filter

Perhitungan nilai komponen adalah sbb:

- frekwensi cut-off : 50 Hz
- dipilih $R = 10 \text{ k}\Omega$

maka besarnya C dapat dihitung dari rumus:

$$f_c = \frac{0,707}{2\pi \cdot C \cdot R} \quad \text{sehingga didapatkan}$$

$$C = 220 \text{ nF}$$

Pada Tugas Akhir ini dibuat tiga buah sensor, yaitu untuk pengukuran pada gigi seri dan pada kedua sisi gigi geraham. Sehingga diperlukan 3 buah Jembatan Wheatstone, masing-masing output diperkuat dengan Instrumentasi Amplifier dan penyaring noise frekwensi tinggi Low Pass Filter 50 Hz.

3.2.3. MULTIPLEXER 4052

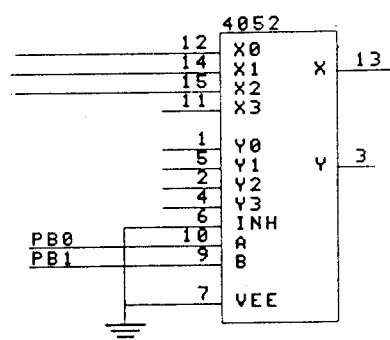
Untuk menyeleksi input mana dari ketiga sensor yang diaktifkan digunakan Multiplexer 4052. Multiplexer (MUX) 4052 ini memiliki 4 buah input dan dalam 1 IC berisi 2 buah MUX. Disini multiplexer ini dipasang pada supply +5 V dan 0 V sehingga input hanya dalam range 0 - 5 V saja, input negatif tidak dilewatkan oleh multiplexer ini.

Input multiplexer ini dikontrol dengan menggunakan input PPI PC0 dan PC1 pada pin A dan B. Karena pada Tugas Akhir ini hanya 3 input saja yang akan diseleksi, maka input yang digunakan hanya 3, yaitu X0, X1 dan X2. Input 1 berasal dari sinyal untuk pengukuran gigi seri sedang input 2 dan 3 berasal dari sinyal untuk pengukuran gigi geraham. Pada masing-masing input diberikan tahanan input 10k Ω sebagai tahanan kompensasi.

Sewaktu dilakukan pengukuran pada gigi seri, input 1 saja yang aktif (X0 terhubung dengan output) dan saat dilakukan pengukuran pada gigi geraham, input 2 dan 3 aktif secara bergantian (PC0 dan PC1 ==> 01 dan 10) berulang-ulang hingga pengukuran selesai dilakukan. Gambar rangkaian Multi-

plexer 4052 ditunjukkan pada Gambar 2.4.

Output hasil seleksi multiplexer 4052 yang masih merupakan besaran analog ini diubah menjadi besaran digital dengan menggunakan Konverter dari analog ke digital.



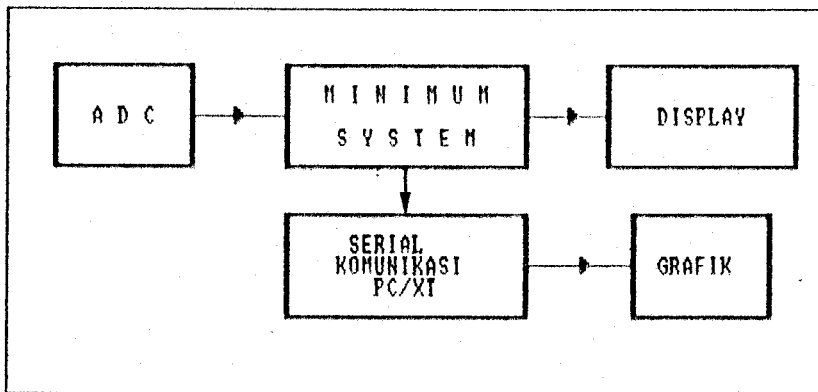
Gambar 3.2.4

Multiplexer 4052

3.3 PENGOLAH SINYAL DIGITAL

Blok diagram pengolah sinyal digital ditunjukkan pada Gambar 3.3.1 dibawah ini.

Komponen-komponen utama yang bekerja pada pengolah sinyal digital ini semuanya dikontrol oleh μP dan dijalankan dengan menggunakan program (software). Dan diinisialisasi sesuai dengan kebutuhan.

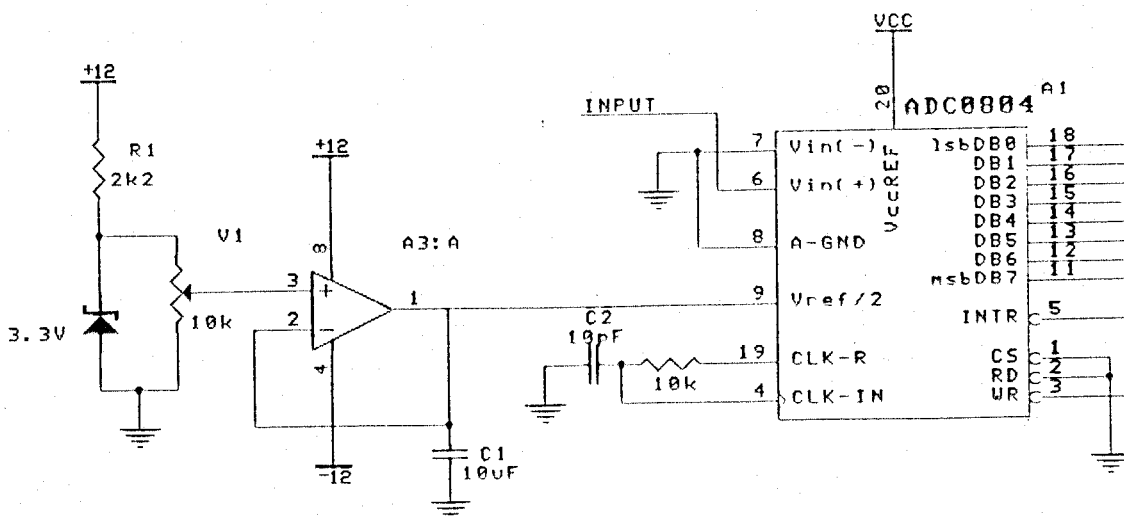


Gambar 3.3.1

Blok Diagram Pengolah Sinyal Digital

3.3.1 KONVERSI ANALOG KE DIGITAL DENGAN ADC 0804

Konverter Analog ke Digital yang direncanakan menggunakan ADC 0804 dengan output 8 bit. Gambar rangkaian ADC 0804 tersebut adalah sebagai berikut:

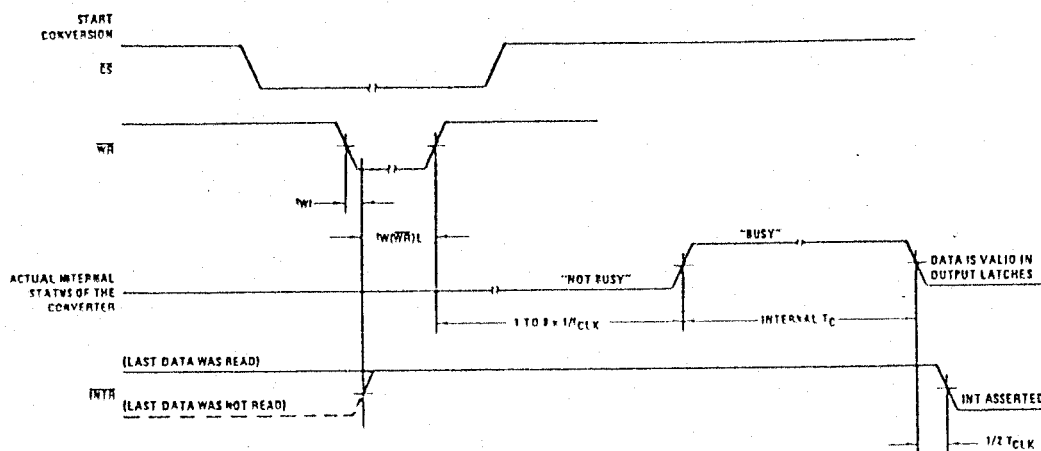


Gambar 3.3.2

Rangkaian ADC 0804

Karena urutan yang sudah tertentu ini meskipun Chip Select ($CS_{\bar{}}$) selalu diaktifkan (diground) tidak akan terjadi kesalahan oleh μP . Sedangkan untuk line controlnya, ADC 0804 ini digunakan Port C dari salah satu PPI 8255, dengan susunan sebagai berikut : $RD_{\bar{}}$ dihubungkan dengan PC2 dan $WR_{\bar{}}$ dengan PC3 sedangkan $INTR_{\bar{}}$ terhubung pada PC4. Sesuai dengan kebutuhan ADC ini, saat inisialisasi PPI dikirim control word yang mengatur Port A sebagai input, Port C lower sebagai output dan Port C upper sebagai input. Port B tidak digunakan pada ADC tetapi digunakan untuk mengontrol tombol.

Timing Diagram ADC 0804 adalah sebagai berikut :



Gambar 3.3.3

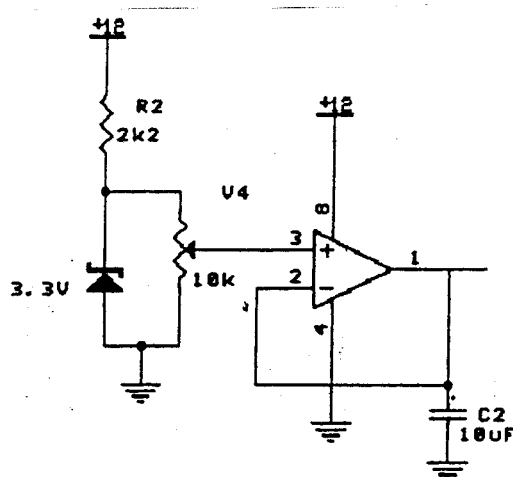
Timing Diagram ADC 0804

Dengan melihat timing diagram ADC 0804, kontrol pada ADC dapat diprogram sesuai timing diagram tsb diatas. Pada keadaan normal $WR_{\bar{}}$ dan $RD_{\bar{}}$ "high" (diberikan melalui PC2 dan PC3). Konversi dimulai pada saat $WR_{\bar{}}$ diberi sinyal "low", sinyal analog yang telah ada pada input (pin 6) diambil dan

dikonversi oleh ADC. Pin INTR_ akan "high" apabila konversi telah selesai, dan data digital 8 bit hasil konversi telah berada di output (pin 10-17) dan telah siap untuk dibaca. Sebelum data diambil RD_ dibuat "low" sehingga INTR_ kembali "low". Setelah itu RD_ dan WR_ diset "high" kembali.

Pengambilan data dilakukan melalui port A PPI, dan data yang didapat dimasukkan ke dalam RAM dan dikonversi menjadi satuan tekanan (kilogram) untuk kemudian ditampilkan pada display. Demikian seterusnya konversi dilakukan diawali dengan memberikan sinyal "low" pada line control WR_ dan RD_.

Direncanakan input maksimum dari ADC 0804 ini adalah 3 Volt, karena itu pada pin 9 sebagai input Vref/2 diberikan tegangan 1,5 Volt. Tegangan ini dihasilkan dari rangkaian referensi menggunakan IC LM 358, sebuah diode zener 3,3V dan 2 buah resistor serta sebuah kapasitor 10 μ F. Rangkaian referensi dapat dilihat pada gambar 3.3.4. dibawah ini.



Gambar 3.3.4

Penghasil Tegangan Referensi 1,5 V

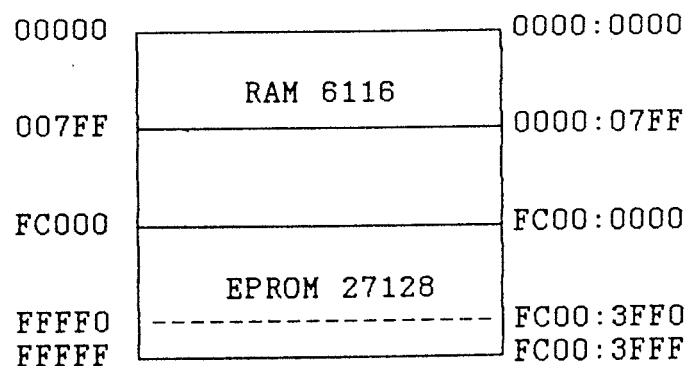
Output dari ADC ini diambil oleh PPI 8255 melalui port A dan hasil konversi disimpan pada RAM dari minimum sistem dan ditampilkan pada display.

3.3.2 CENTRAL PROCESSING UNIT MINIMUM SISTEM 8088

Sebagai prosessor pengendali digunakan minimum sistem 8088 yang dilengkapi dengan sebuah RAM dan sebuah ROM. Memory yang direncanakan menggunakan satu RAM statis 6116 dan satu EPROM 27128. Memory mapping yang direncanakan dapat dilihat pada gambar 3.3.5.

EPROM diletakkan pada address dengan lokasi tertinggi, hal ini dilakukan dengan alasan setelah terjadi reset pada μP 8088, mikroprosesor akan menjalankan program pada physical address FFFF0h. Pada lokasi memory FFFF0 diletakkan instruksi jump far ke awal program sehingga program yang telah direncanakan dapat dijalankan.

Program yang direncanakan diletakkan pada EPROM dengan lokasi physical address FC000h - FFFFFh (16 kbyte). Sedangkan lokasi 00000h - 007FFh (2 kbyte) digunakan sebagai RAM untuk data dan stack.



Gambar 3.3.5. Memory map

Interval memory dan zone bit dari RAM 6116 dan EPROM 27128, bila digunakan sistem full decoding, maka seluruh zone bit ikut diperhitungkan.

PPI 8255, PIC 8259 dan PIT 8253 ketiganya dikontrol melalui CPU ini dan pengalamatannya juga saling disesuaikan satu sama lain sehingga tidak ada alamat yang sama. Semua proses dilaksanakan sesuai dengan program yang disimpan pada EPROM.

3.3.3 RANGKAIAN TIMER DAN INTERRUPT

Pengukuran kekuatan gigit ini dilakukan selama 3 detik, dan setelah ini hasil yang tercetak pada display adalah nilai tertinggi dari kekuatan gigit pasien tersebut. Sebagai pengendali waktu digunakan rangkaian Timer dan Interrupt ini. Rangkaian Timer berfungsi sebagai penghitung waktu, sedangkan Rangkaian Interrupt berfungsi untuk menghentikan proses setelah waktu tersebut dipenuhi.

Timer yang direncanakan ini menggunakan *Programmable Interval Timer* (PIT) 8253 dan untuk interruptnya digunakan *Priority Interrupt Controller* (PIC) 8259, yang masing-masing memakai alamat 004h-007h dan 008h-009h.

Timer dan Interrupt ini dijalankan dengan program. Pada awal program masing-masing timer dan interrupt diinisialisasi terlebih dahulu untuk menentukan mode ataupun ketentuan lain yang diinginkan.

Rangkaian ini juga dilengkapi IC 74LS393 sebagai pembagi frekwensi, pada rancangan ini dipilih frekwensi 20 Hertz

sebagai input clock pada PIT 8253.

Pada PIT 8259 dikirimkan control word 00111110b, yang berarti dipilih counter0, membaca/menulis *Least Significant Byte* (LSB) lebih dulu, baru kemudian *Most Significant Byte* (MSB), mode ke 3 yang berarti memilih *square wave generator* (menghasilkan pulsa kotak) dan terakhir diinginkan binary counter 16 bit.

Kemudian pada alamat Counter0 dikirimkan nilai pembagi 29818 (3A3Ah) agar didapatkan frekwensi 10 Hz. Cara pengiriman, dilakukan dengan mengirim LSB lebih dahulu ke alamat counter0 baru MSBnya dikirim kemudian.

Sedangkan pada PIC 8259 dikirimkan ICW1 00010011b, yang artinya ICW4 dibutuhkan, single bukan cascade, address interval 8 dan *edge triggered mode*. Pada ICW2 dikirimkan alamat interrupt vektor dari PIC ini yaitu 08h.

Pada ICW4 dikirimkan control word 00001001b, yang berarti IRQ0 - IRQ7 memiliki prioritas yang sama, normal *End Of Interrupt* (EOI) serta jenis μP adalah 8088. Sedangkan untuk control word OCW1 dikirimkan data 11111110b, yang berarti IRQ2 - IRQ7 tidak digunakan (diberi mask).

Program untuk rangkaian Timer dan Interrupt diatur melalui procedure program. Setiap kita memasukkan nilai pada memory counter secara otomatis procedure ini langsung dijalankan. Dari rangkaian dan program dihasilkan penghitung dengan prinsip sbb. Timer 8253 memberikan clock pada PIC 8259, dan pada setiap perubahan clock PIC akan menginterrupt Timer sehingga menghitung sesuai banyaknya interrupt. Diinginkan penekanan dilakukan selama 3 detik, sedangkan frek-

wensi timer ini 20 Hz, berarti saat menghitung hingga hitungan ke 60 loop pengambilan data dihentikan.

3.3.4. PROGRAMMABLE PERIPHERAL INTERFACE 8255

Pada pengerjaan Tugas Akhir ini digunakan 2 buah PPI 8555, sebuah PPI dengan alamat 000h-003h digunakan untuk mengambil input dari ADC 0804 (Port A), sebagai kontrol input Multiplexer 4052 (PC0 dan PC1), sebagai kontrol pada ADC (PC2, PC3 dan PC4) dan sebagai pengontrol tombol (push-button switch) yaitu pada Port B. Sehingga pada alamat control word pada PPI ini diberikan data 10011010b, yang berarti PPI diset pada mode 0, Port A, Port B dan port C upper sebagai input sedang Port C lower sebagai output.

Sedangkan PPI yang lain dengan alamat 00Ch-00Fh digunakan untuk mengendalikan display LCD (Liquid Crystal Display). Pada alamat control word PPI ini dikirimkan data 80h, yang berarti PPI bekerja pada mode 0 dan semua port difungsikan sebagai output. Port C pada PPI ini tidak digunakan.

3.3.5 PERAGA (DISPLAY DAN TOMBOL)

Tombol dan display yang akan dipakai pada Tugas Akhir ini dikontrol menggunakan PPI 8255.

a> Tombol (pushbutton switch)

Tombol yang diperlukan pada peraga adalah 6 buah, tetapi karena tombol ini dikontrol atau dicek melalui port B PPI 8255 disediakan 8 buah tombol, 2 yang lain bisa di-

gunakan jika diperlukan. Tombol ini dirangkai *pull up* menggunakan resistor 4,7k Ω dan Vcc 5 Volt sisi yang lain dihubungkan pada ground. Tombol ini dicek setiap saat oleh μ P melalui port B PPI. Dalam keadaan tidak ada tombol yang ditekan, input yang masuk ke Port B semua "high" dan apabila ada satu tombol yang ditekan, maka rangkaian akan langsung short ke ground, sehingga port yang tersambung dengan tombol tersebut akan mendapatkan input "low". Dan μ P akan menjalankan program sesuai posisi tombol tersebut. Fungsi masing-masing tombol dapat dilihat sbb:

1. Tombol START terhubung dengan PBO -> apabila tombol ini ditekan alat telah siap digunakan dan pada display akan tampak tulisan 'READY'.
2. Tombol KALIBRASI terhubung dengan PB1 -> apabila tombol ini ditekan, maka dilakukan kalibrasi pada ADC dengan cara semua pengukuran yang dilakukan kemudian hasilnya akan dikurang input ADC saat itu. Ini dilakukan agar apabila plat tidak kembali normal setelah terjadi penekanan sehingga output dari jembatan Wheatsone tidak nol.
3. Tombol GIGISERI terhubung dengan PB2 -> apabila tombol ini ditekan maka multiplexer akan mengaktifkan input 1, yang terhubung dengan alat untuk mengukur kekuatan gigit pada gigi seri.
4. Tombol GERAHAM terhubung dengan PB3 -> apabila tombol ini ditekan maka multiplexer akan mengaktifkan input 2 dan 3 secara bergantian. Input 2 terhubung dengan

sisi plat sebelah kanan (untuk mengukur kekuatan gigit pada geraham kanan), sedangkan input 3 terhubung dengan sisi plat sebelah kiri (untuk mengukur kekuatan gigit pada geraham kiri).

5. Tombol CLEAR terhubung dengan PB6 -> apabila tombol ini ditekan, maka semua data yang ada pada memory (RAM) dihilangkan.
6. Tombol KIRIM DATA terhubung dengan PB7 -> apabila tombol ini ditekan, maka data hasil konversi yang ada pada RAM dikirimkan ke CPU PC dan hasilnya akan ditampilkan sebagai grafik.

b> Liquid Crystal Display (LCD)

Display yang digunakan untuk menunjukkan besarnya hasil pengukuran pada Tugas Akhir ini adalah LCD. LCD ini memiliki 14 buah pin, yaitu 2 pin untuk power (+ 5V dan 0V), 4 buah pin kontrol : RD/WR_, Vss untuk kontras (maksimum 0V), RS dan E (Enable) serta 8 buah pin untuk 8 bit data.

LCD ini memiliki banyak sekali parameter. Sesuai dengan kebutuhan pada Tugas Akhir ini, melalui inisialisasi LCD diinginkan display dengan 2 line (16 karakter), masing-masing karakter 5x7 dot matrix. Kursor dibuat tidak tampak dan display dibuat tidak bergeser.

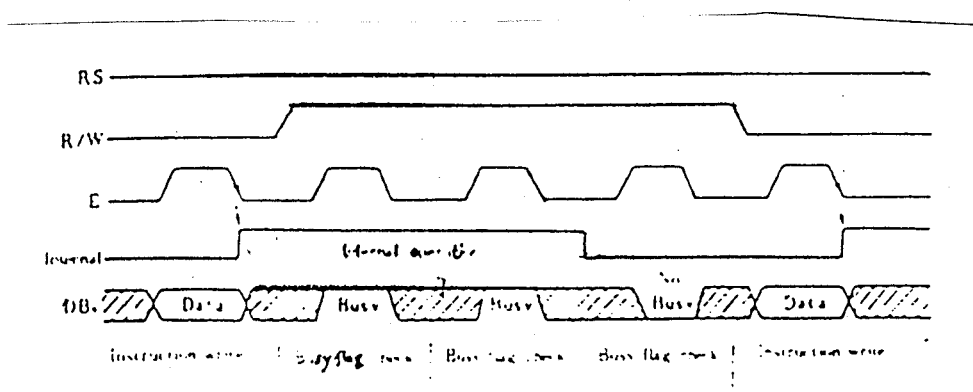
Display ini dikontrol menggunakan PPI 8255, 8 bit data dikontrol melalui port A, dan 4 pin kontrol LCD yang lain dikontrol melalui port B.

Vss atau kontras dari LCD diberikan input '0', sehingga

kontras selalu maksimum. Saat dilakukan pengiriman data input RD/WR_ diset '0', sedangkan RS digunakan untuk mengetahui apakah data yang dikirim berupa instruksi atau data karakter. RS '0' untuk pengiriman instruksi dan RS '1' untuk pengiriman data karakter.

Data yang dimasukkan berupa data Ascii. Timing diagram dari LCD ini dapat dilihat pada Gambar 6.6.5. Dari Timing diagram dapat dilihat bahwa Enable aktif setiap ada perubahan dari "high" ke "low". Sehingga setiap pengiriman data selalu diikuti dengan perubahan enable ini.

Output tegangan dari ADC yang kemudian dikonversikan dalam satuan kilogram, ditampilkan pada LCD ini. Untuk pengukuran pada gigi seri, hasil pengukuran ditampilkan pada line pertama saja, sedangkan pada pengukuran gigi geraham, hasil pengukuran untuk geraham sebelah kiri ditampilkan pada line pertama sedangkan hasil pengukuran pada geraham kanan ditampilkan pada line kedua.



Gambar 3.3.6

Timing Diagram LCD

3.4. SERIAL KOMUNIKASI INTERFACE

Serial komunikasi yang direncanakan ini menggunakan card serial yang kompatibel dengan IBM Personal Computer, yaitu RS-232. Pada Tugas Akhir ini, serial komunikasi digunakan untuk menampilkan data hasil pengukuran dalam bentuk grafik.

Data yang dikirimkan adalah data hasil konversi dari output ADC yang berupa tegangan menjadi besaran beban dalam satuan kilogram.

Pada inisialisasi RS-232 diatur alamat-alamat port yang dipilih, disini dipilih bagian COM1 yaitu dari 03F8h - 03FFh. secara lengkap yaitu sbb:

Tx Buffer	-	03F8H
Rx Buffer	-	03F8H
Baudrate Divisor Latch LSB	-	03F8H
Baudrate Divisor Latch MSB	-	03F9H
Interrupt Enable Register	-	03F9H
Interrupt Identification Register	-	03FAH
Line Control Register	-	03FBH
Modem Control Register	-	03FCH
Line Status Register	-	03FDH
Modem Status Register	-	03FEH

Pada alamat Line Control Register dikirimkan data 10000000b, bit ke-7 diberi '1' untuk *memilih baud rate divisor*. Kemudian dikirimkan *baud rate divisor 300 (0108h)* pada alamat Baudrate Divisor Latch, yang dikirimkan MSBnya lebih dahulu. Pada alamat Line Control Register dikirimkan kembali data

00000011b, yang berarti diinginkan pengiriman *8 bit data*, *1 stop bit* dan *parity disable*. Dan yang terakhir dikirimkan data 00h ke alamat Interrupt enable register yang berarti *interrupt disable*.

Pada pengiriman data serial ini diberikan batasan sbb bahwa 10h adalah karakter kontrol awal data pada pengukuran gigi seri dan 11h adalah karakter kontrol akhirnya, sedangkan 12h dan 13h adalah karakter kontrol bagi data pengukuran gigi geraham.

Data yang dikirimkan melalui serial komunikasi ini adalah output ADC yang telah dikonversikan dalam mV dan dikirim hasil konversi ke desimal. Sebelum pengiriman data selalu dilakukan pengecekan ini Tx Buffer, pengecekan ini dilakukan dengan cara mengambil data yang ada pada line control register, bit ke-5 menunjukkan kondisi Tx buffer ('1' berarti Tx buffer sudah kosong). Setelah mengetahui bahwa Tx buffer telah kosong, baru data mulai dikirimkan dan selalu diawali dengan pengiriman karakter kontrol awal dan diakhiri karakter kontrol akhir. data yang berada diantaranya inilah data yang sesungguhnya. Sebelum pengiriman, data yang tadinya berupa biner(integer) diubah lebih dahulu menjadi desimal dan yang dikirim adalah nilai ASCII-nya, hal ini dilakukan sebagai pengaman saja, daripada yang dikirim langsung adalah data dalam biner tsb.

Pengiriman data serial ini dilakukan setelah dilakukan penekanan tombol 'KIRIM DATA'. Data yang dikirim adalah data yang berada pada memory out_adc yang telah dikonversi. Jadi pengiriman data selalu setelah dilakukan pengukuran, sehing-

ga pada memory ini telah ada datanya. Apabila sebelum ditekan tombol 'KIRIM DATA' dilakukan pengukuran gigiseri, maka data yang dikirimkan kemudian adalah data hasil pengukuran gigiseri. Demikian juga bila sebelumnya dilakukan pengukuran pada gigi geraham.

Pada pengukuran gigi geraham hasil pengukuran disimpan pada memory bergantian kiri dan kanan, sehingga saat pengiriman data serial pengolahan data juga diambil bergantian. Sedangkan pada pengukuran gigiseri, data yang dikirimkan diselingi dengan pengiriman angka nol, hal ini dilakukan agar pengolahan datanya sama dengan jika dilakukan pengukuran gigi geraham.

Hasil komunikasi serial ini diolah menggunakan bahasa pemrograman tingkat tinggi, yaitu bahasa C. Hal ini untuk mempermudah pengolahan data untuk ditampilkan dalam bentuk grafik. Pada pengukuran gigi geraham, untuk hasil kiri dan kanan ditampilkan dalam satu koordinat dengan warna yang berbeda, sedangkan pada pengukuran gigiseri dalam satu warna dan warna yang lain dinolkan.

BAB IV

KALIBRASI DAN PENGUKURAN

Untuk mengetahui hasil dari pembuatan alat ukur ini, telah dilakukan langkah-langkah agar didapatkan hasil yang paling sesuai dan benar sesuai dengan keadaan yang sesungguhnya. Karena itu dilakukan kalibrasi dan uji ukur, yang dibahas pada bab ini.

4.1 KALIBRASI

Sebelum peralatan ini digunakan untuk mendapatkan data yang baik maka dilakukan kalibrasi alat terlebih dahulu. Bagian-bagian yang perlu dilakukan kalibrasi adalah:

1. Jembatan Wheatstone

Untuk dua macam pengukuran yang dilakukan, yaitu pada gigiseri dan gigi geraham, dibutuhkan tiga buah jembatan Wheatstone yang masing-masing dilengkapi dengan penyeimbang jembatan. Kalibrasi atau penyeimbangan jembatan wheatsone ini dilakukan dengan mengatur tahanan variabel VR1 (Gambar 3.2.1) sehingga output jembatan saat tidak ada gerakan adalah 0 mV.

2. Penguat Instrumentasi

Sebagai penguat instrumentasi (Instrumentasi Amplifier) digunakan IC AD521JD. Offset nol pada penguat instrumentasi ini dilakukan dengan mengatur tahanan variabel VR2 (Gambar 3.2.2) sehingga output penguat instrumentasi ini 0 mV saat kedua inputnya nol.

4.2. PENGUKURAN

Pengukuran atau pengambilan data yang dilakukan terdiri dari 3 bagian, yaitu:

- * Output Instrumentasi Amplifier
- * Output Konverter Analog ke Digital
- * Output Hasil Pembebanan

4.2.2. Output Instrumentasi Amplifier

Pada pembuatan alat ukur ini, digunakan 3 buah penguat instrumentasi, untuk mengatur penguatan ketiga buah sensor. Dengan mengatur tahanan variabel VR3 (Gambar 3.2.2) diset penguatan sebesar 500 kali. Hasil test pengukuran penguat instrumentasi ini dapat dilihat pada tabel 4.1, 4.2 dan 4.3.

Tabel 4.1

Hasil Pengukuran Penguat Instrumentasi untuk Pengukuran Gigiseri

Teg Input (mV)	Teg Output (V)	Error
1,0	0.500	0
2,0	0.985	0.015
3,0	1.493	0.007
4,0	1.982	0.012
5,0	2.455	0.015
6,0	2.942	0.058
7,0	3.41	0.09
8,0	3.9	0.1

Tabel 4.2

Hasil Pengukuran Penguat Instrumentasi untuk Pengukuran Gigi Geraham Kanan

Teg Input (mV)	Teg Output (V)	Error
1,0	0.47	0.03
2,0	1.02	0.02
3,0	1.51	0.01
4,0	2.00	0.00
5,0	2.520	0.02
6,0	3.02	0.02
7,0	3.55	0.05
8,0	4.04	0.04

Tabel 4.3

Hasil Pengukuran Penguat Instrumentasi untuk Pengukuran Gigi Geraham Kiri

Teg Input (mV)	Teg Output (V)	Error
1,0	0.478	0.022
2,0	1.015	0.015
3,0	1.521	0.021
4,0	2.00	0.00
5,0	2.541	0.041
6,0	3.02	0.02
7,0	3.52	0.02
8,0	4.04	0.04

Dari hasil pengukuran tersebut diatas, maka didapatkan grafik hubungan tegangan input dan output dari penguat instrumentasi. Dan dari hasil tersebut dapat dihitung regresi linier dari masing-masing penguat, yaitu:

* Pengukuran gigiseri:

$$V_o = 0.0256 + 0.485 V_i$$

penguatannya menjadi : 485 kali

* Pengukuran gigi geraham kanan:

$$V_o = - 0.02071 + 0.508 V_i$$

penguatannya menjadi : 508 kali

* Pengukuran gigi geraham kiri:

$$V_o = 0,00975 + 0,506 V_i$$

penguatannya menjadi : 506 kali

Dari hasil perhitungan dengan menggunakan cara regresi linier, terlihat kelinieran masing-masing penguat instrumentasi tersebut diatas. Perubahan nilai penguatan ini disebabkan antara lain karena:

- penguat instrumentasi ini mempunyai kemampuan penguatan yang tinggi, sehingga ketika dilakukan kalibrasi kedua input penguat adalah 0 mV, tegangan sesungguhnya masih ada dengan ketelitian perseratus atau perseribu (kemampuan maksimum AD521JD hingga penguatan 1000 kali). Hal ini tidak bisa dideteksi karena alat ukur yang digunakan mempunyai ketelitian hanya hingga sepersepuluh saja.
- karakteristik dari masing-masing komponen.
- pemilihan tegangan referensi input yang tidak sama

4.4.3. Output Hasil Pembebanan

Tegangan keluaran penguat instrumentasi diukur dengan

memberikan tekanan tertentu. Tes Pembebanan ini dilakukan di Laboratorium Dasar Bersama Universitas Airlangga Surabaya dengan menggunakan alat pembebanan *Autograph*. Pengujian dilakukan bergantian dari ketiga alat ukur dengan kenaikan gaya setiap 50 Newton. Gaya 50 Newton ini setara dengan 5,1 kg (Gaya tekan dibagi gravitasi).

Hasil pengukuran dari alat ukur ini dapat dilihat pada Tabel 4.4, Tabel 4.5 dan Tabel 4.6.

Tabel 4.4

Data Gaya pada Alat Pengukur Kekuatan Gigiseri

Gaya (N)	Hasil pengukuran (mV)
50	0,4
100	0,8
150	1,3
200	1,7
250	2,1

Tabel 4.5

Data Gaya pada Alat Pengukur Kekuatan Gigi Geraham Kanan

Gaya (N)	Hasil pengukuran (mV)
50	0,6
100	1,2
150	1,9
200	2,7
250	3,4

Tabel 4.6

Data Gaya pada Alat Pengukur Kekuatan Gigi Geraham Kiri

Gaya (N)	Hasil pengukuran (mV)
50	0,6
100	1,2
150	2,0
200	2,7
250	3,5

Dari ketiga data hasil pengukuran, maka didapatkan regresi linier dari masing-masing pengukuran, yaitu:

* Pengukuran gigiseri:

$$y = - 0.03 + 0.0086 x$$

perubahan tegangan untuk setiap Newton perubahan tekanan besarnya: 0,0086 mV atau 0,86 mV setiap kenaikan tekanan 100 N (± 10 kg).

* Pengukuran gigi geraham kanan:

$$y = - 0.17 + 0.0142 x$$

perubahan tegangan untuk setiap Newton perubahan tekanan besarnya: 0,0142 mV atau 1,42 mV setiap kenaikan tekanan 100 N (± 10 kg).

* Pengukuran gigi geraham kiri:

$$y = - 0.19 + 0.0146 x$$

perubahan tegangan untuk setiap Newton perubahan tekanan besarnya: 0,0146 mV atau 1,46 mV setiap kenaikan

tekanan 100 N (± 10 kg).

Pada test pembebanan ini beban diberikan dalam satuan Newton. Konversi dari gaya beban dari newton ke kilogram ini telah diterangkan diatas yaitu 100 N sebanding dengan 10,2 kg.

Dari ketiga jenis sensor ini, yaitu untuk gigiseri, gigi geraham kanan dan kiri, terdapat perbedaan linearitas yang jelas dari sensor pada gigiseri dan pada gigi geraham. Hal ini disebabkan karena jenis strain gauge yang digunakan pada gigiseri lain dengan strain gauge yang digunakan pada sensor gigi geraham kanan dan kiri sehingga faktor gauganya juga lain. Sedangkan untuk sensor pada gigi geraham kiri dan kanan juga ada perbedaan kecil untuk linearitasnya sedangkan jenis dari strain gauge ini sama. Hal ini dikarenakan pada cara pemasangannya, ini sangat mungkin terjadi karena pemasang strain gauge ini bukan orang yang ahli dalam hal tersebut.

4.4.4. Hasil Pengukuran ADC

Input dari rangkaian ADC 0804 merupakan keluaran dari penguat instrumentasi. Sesuai dengan konstanta linearitas dari strain gauge dan keterbatasan kemampuan plat logam, maka kekuatan tekan pada plat maksimum 50 kg. Tekanan 50 kg pada plat sebanding dengan tegangan 2 Volt pada plat pengukur kekuatan gigiseri dan 3 Volt pada plat pengukur kekuatan gigi geraham. Karenanya tegangan referensi diatur setengah

kali tegangan input maksimum, sehingga tegangan referensinya menjadi 1,5 Volt.

ADC 0804 adalah konverter 8 bit, sehingga data biner maksimum adalah 256. Karenanya untuk input maksimum 3 Volt, perubahan tegangan setiap bitnya adalah

$$\frac{3}{256} = 0.0117 \text{ V} = 11.7 \text{ mV}$$

Jadi setiap kenaikan tegangan 11.7 mV pada input ADC, pada output ADC naik 1 bit.

Konversi hasil pengukuran yaitu output tegangan jembatan Wheatstone yang kemudian diperkuat dengan penguat instrumentasi sebesar kurang lebih 500 kali, dimana hasil penguatan yang berupa sinyal analog ini dikonversi menjadi nilai digital oleh ADC 0804 dengan kenaikan per bit-nya 11.7 mV. Untuk mendapatkan nilai kekuatan gigit dalam skala kilogram, maka output ADC yang berupa besarnya tegangan harus dikonversikan dengan mengalikan dengan konstanta pengali sesuai dengan besarnya perubahan tegangan masing-masing sensor untuk perubahan beban setiap kilogram.

Perhitungan konstanta konversi adalah sbb:

$$x = y \times 11,7 : z$$

dimana x = hasil konversi dalam kg
 y = output adc dalam biner (0-255)
 z = besarnya tegangan saat diberi beban x (saat test pembebanan)
 konstanta pengali = 11,7 : z

Karena harus integer, konstanta pengali diberikan dengan dikalikan kelipatan sepuluh seperlunya, nanti saat hasil ditampilkan pada display kita tinggal mengatur dimana posisi titik desimal diletakkan. Pada perhitungan ini dikalikan dengan angka 1000, sehingga nanti pada hasil perhitungannya tiga angka terakhir adalah desimal.

Untuk pengukuran pada gigi seri, didapatkan konstanta pengali 292 dan untuk pengukuran gigi geraham, keduanya memakai konstanta pengali 195. Hasil perkalian ini dalam 3 tempat desimal.

Konversi ini dilakukan oleh μP dengan operasi aritmatika yang pelaksanaannya diatur melalui program. Bahasa pemrogram yang digunakan adalah bahasa assembly. Semua operasi pada assembly selalu dalam bilangan integer, karenanya hasil yang juga integer ini, pada saat akan ditampilkan kita atur dimana kita harus meletakkan titik desimal.

Pengambilan input ADC dilakukan setiap 50 milisecond sekali, hal ini cukup memenuhi mengingat waktu berhenti saat mengunyah adalah 194 ms dan waktu satu siklus pengunyahan adalah 0,624 s. Perhitungan waktu ini diperlukan apabila alat ukur ini digunakan untuk mengukur kekuatan kunyah, sedangkan pada pengukuran kekuatan gigit hal ini tidak terlalu berpengaruh.

Dalam satu kali pengukuran dilakukan 60 kali pengambilan data dan output dari ADC yang disimpan dalam RAM ini kemudian dikirimkan melalui serial komunikasi interface ke

suatu PC dan ditampilkan dalam bentuk grafik. Grafik ini adalah grafik besarnya kekuatan gigit sebagai fungsi waktu.

Pada orang sehat kekuatan gigit makin lama akan makin besar hingga maksimum dan akan terus stabil hingga kekuatan otot melemah dan kemudian besarnya kekuatan gigit ini akan makin berkurang. Sedangkan pada orang yang sakit, pada grafik akan tampak bahwa besarnya kekuatan gigit akan naik hingga suatu saat akan langsung turun. Pada puncak kekuatan gigit inilah orang tersebut tidak kuat menahan sakitnya yang mungkin disebabkan ada kerusakan pada oto ataupun syarafnya.

BAB V

PENUTUP

5.1 KESIMPULAN

Dari pembuatan Tugas Akhir ini didapatkan kesimpulan sebagai berikut:

1. Pemakaian sensor strain gauge pada tugas akhir ini karena strain gauge adalah sensor yang digunakan sebagai pendeteksi regangan.
2. Sesuai dengan fungsi pemakaian alat ukur ini, yang digunakan untuk mengukur kekuatan gigit pada manusia, maka alat ini dirancang menggunakan sistem *bending*.
3. Penguat instrumentasi AD251JD memenuhi syarat yang dibutuhkan sebagai penguat input kecil, karena memiliki CMRR yang tinggi, impedansi input besar, impedansi output yang rendah serta linearitas yang tinggi.
4. Komunikasi data serial menggunakan RS-232 merupakan suatu sistem komunikasi data yang cukup sederhana dan dapat memenuhi kebutuhan komunikasi data untuk pembuatan grafik pada alat ukur ini.
5. Alat pengukur kekuatan gigit ini memiliki kemampuan beban maksimum 50 kg.

5.2 SARAN - SARAN

Alat pengukur kekuatan gigit yang dibuat pada Tugas Akhir ini, masih jauh dari sempurna. Dibawah ini diberikan be-

berapa saran-saran atas kekurangan dari Tugas Akhir ini dan diharapkan dapat berguna bagi penelitian dan pengembangan yang lebih lanjut dari tugas akhir ini.

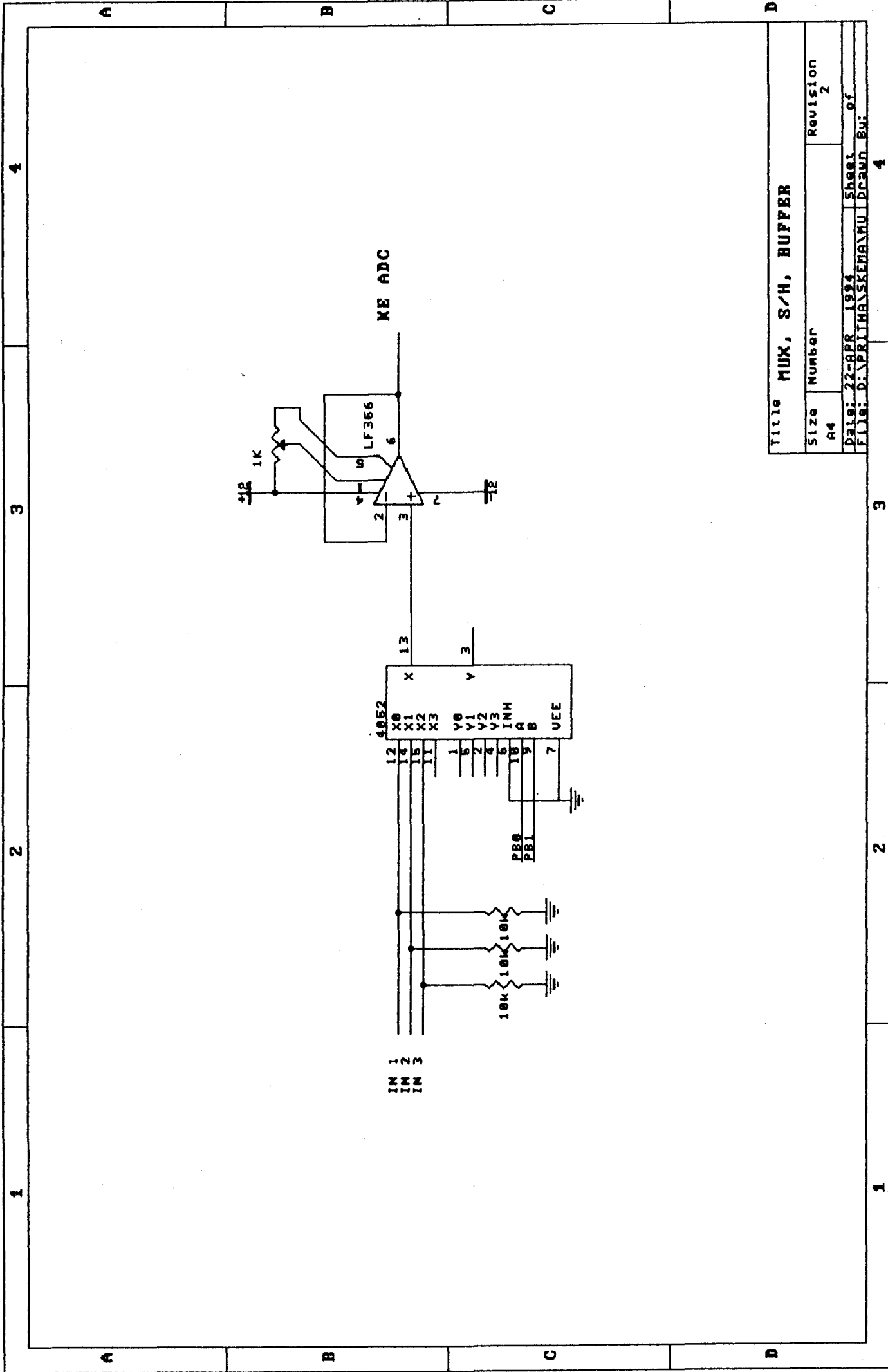
1. Plat logam yang dipilih sebagai sensor ataupun sebagai alas sebaiknya setipis mungkin, dengan elastisitas yang cukup tinggi sehingga mampu dibebani lebih tinggi. Logam yang dipilih tidak boleh dari jenis logam yang dapat menyebabkan kanker.
2. Semakin besar faktor gauge dari strain gauge yang digunakan sebagai transducer, sensor ini akan lebih sensitif.
3. Agar didapatkan hasil pengukuran yang akurat, kalibrasi dan pengukuran hendaknya dilakukan menggunakan alat ukur dengan ketelitian sekecil mungkin.
4. Pada pemakaian yang sesungguhnya sensor alat ukur ini harus dibungkus dengan bahan yang kedap air serta mudah diganti mengingat pemakainya banyak orang. Karena pemakaian alat ini digunakan dalam mulut, sterilitasnya harus dijaga.
5. Alat ini juga dapat dihubungkan dengan printer dan dengan menambah program hasilnya dapat dicetak pada printer.

Demikian kesimpulan dan saran dari pembuatan Tugas Akhir yang berjudul **Alat Pengukur Kekuatan Rahang pada Pemakaian Gigi Tiruan Lengkap yang dapat Diinterfacekan pada PC.**

DAFTAR PUSTAKA

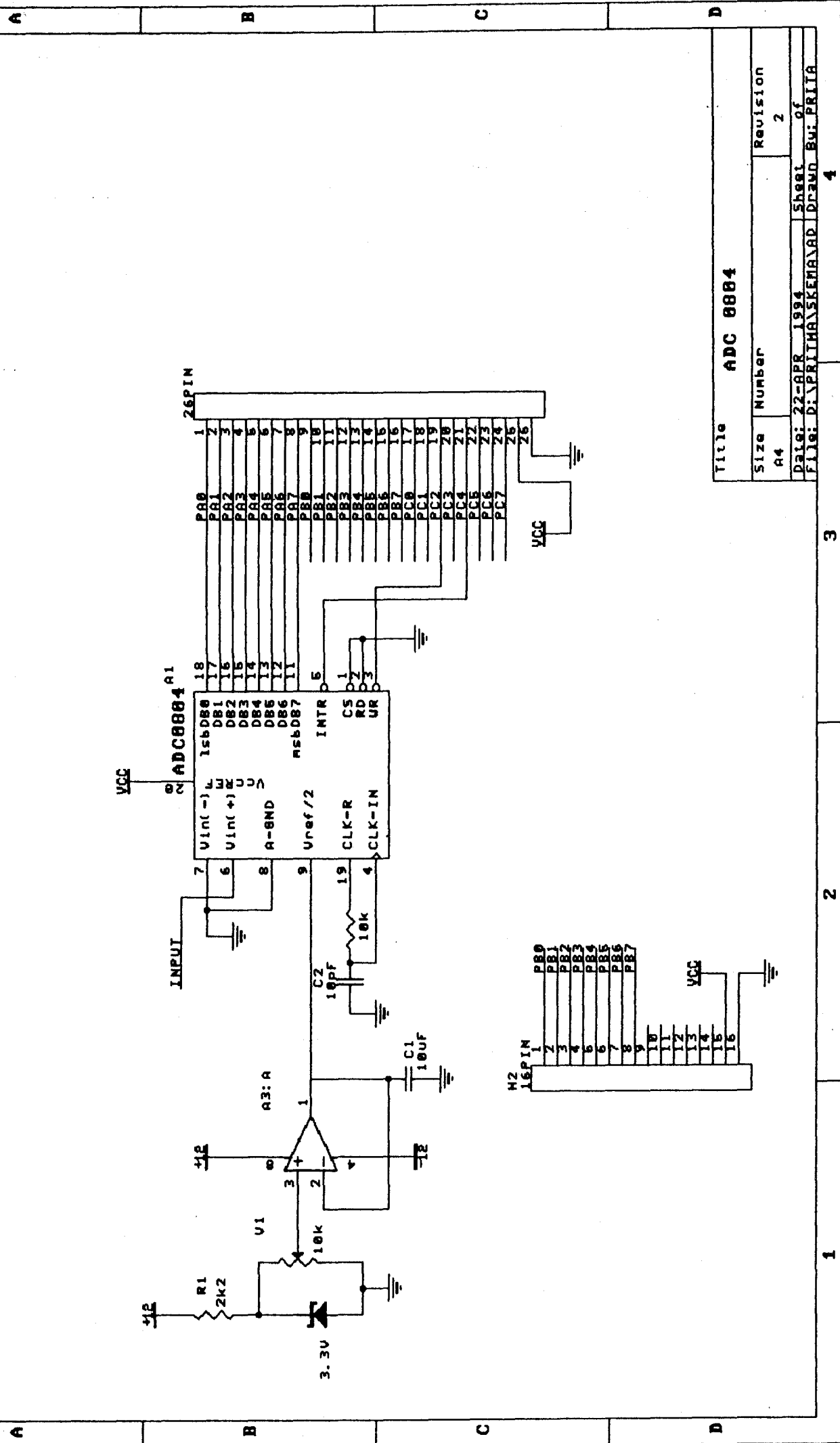
1. Sharry, JJ, **COMPLETE DENTURE PROSTHODONTICS**, Second Edition, The Blakiston Division McGraw Hill Book Co, New York, 1986.
2. Moeljono, dr & Bambang Budi Soesetyo, dr, **DIKTAT ILMU FAAL II**, Universitas Airlangga, Surabaya.
3. Hall, Douglas V, **MICROPROCESSORS AND INTERFACING : Programming and Hardware**, McGraw Hill Book, Co., Singapore, 1986
4. Coughlin, Robert F. & Driscoll, Frederick F., **OPERATIONAL AND LINEAR INTEGRATED CIRCUITS**, Second Edition, Prentice-Hall Inc., Englewood Cliffs, 1982.
5. Uffenbeck, John, **THE 8088/8086 FAMILY: DESIGN, PROGRAMMING AND INTERFACING**, Prentice-Hall Inc., Englewood Cliffs, 1988.
6. Beckwith, Thomas G., Buck, N Lewis & Marangoni, Roy D., **PENGUKURAN MEKANIS**, Edisi Ketiga, Penerbit Erlangga, 1987.
7. Partodarsono, Hartono., **Tuntunan Praktis BAHASA ASSEMBLY**, Gramedia, Jakarta, 1990

8. _____, LINEAR DATABOOK, National Semiconductor Corporation., Santa Clara, 1982
9. _____, MICROPROCESSOR AND PERIPHERAL HANDBOOK
Vol I. Microprocessor, Intel Corporation, 1988.



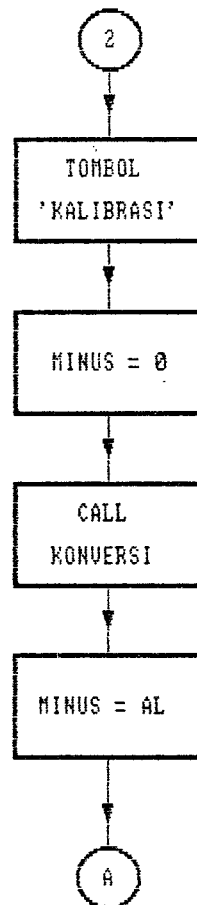
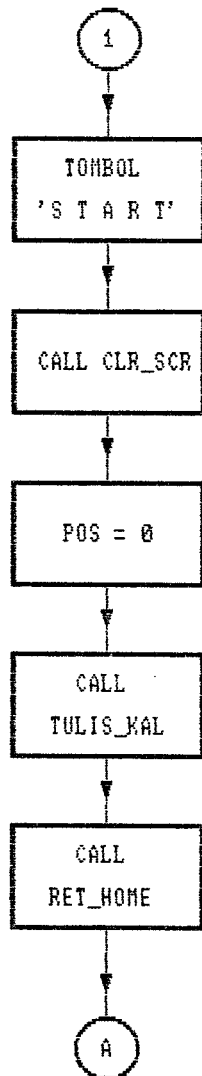
Title MUX, S/H, BUFFER

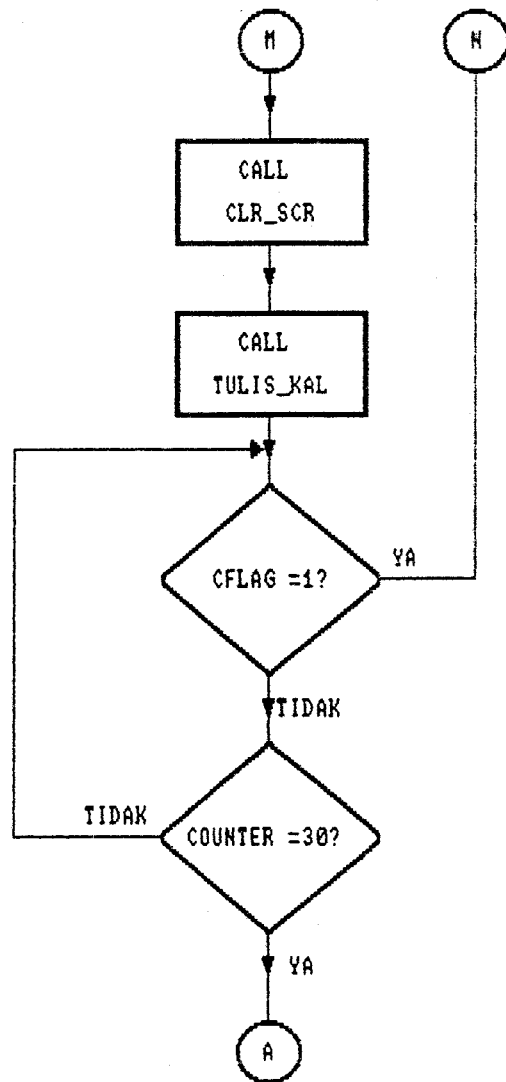
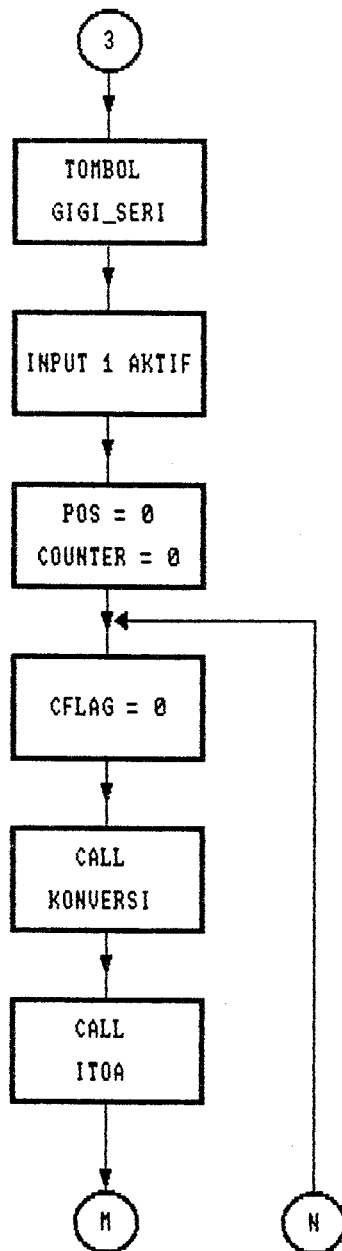
Size	Number	Revision
A4		2
Date: 22-APR-1994 Sheet of		
File: D:\PRATHA\SEKHA\MU Drawn By:		

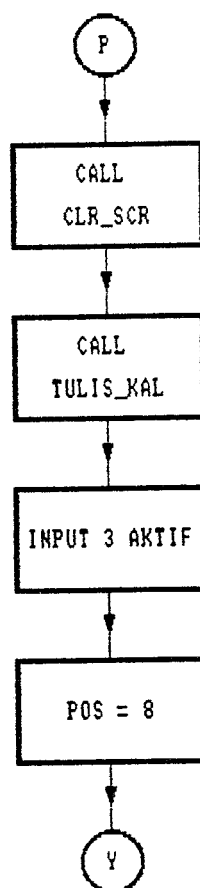
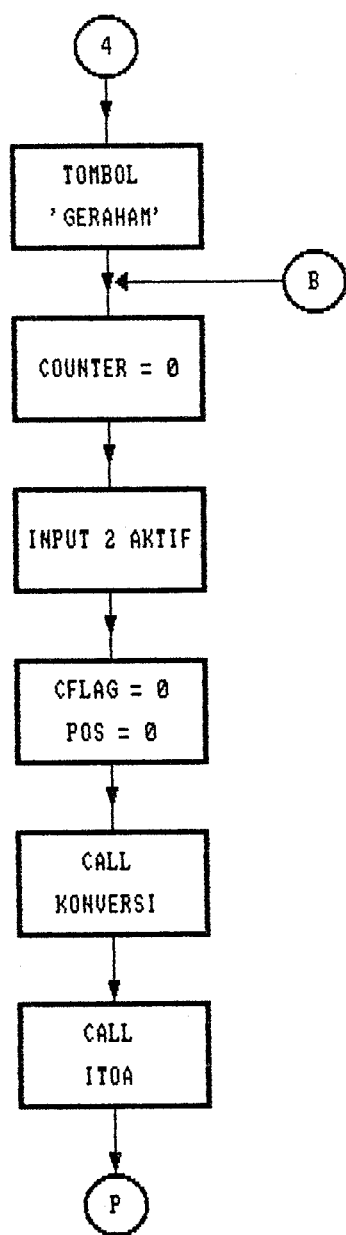


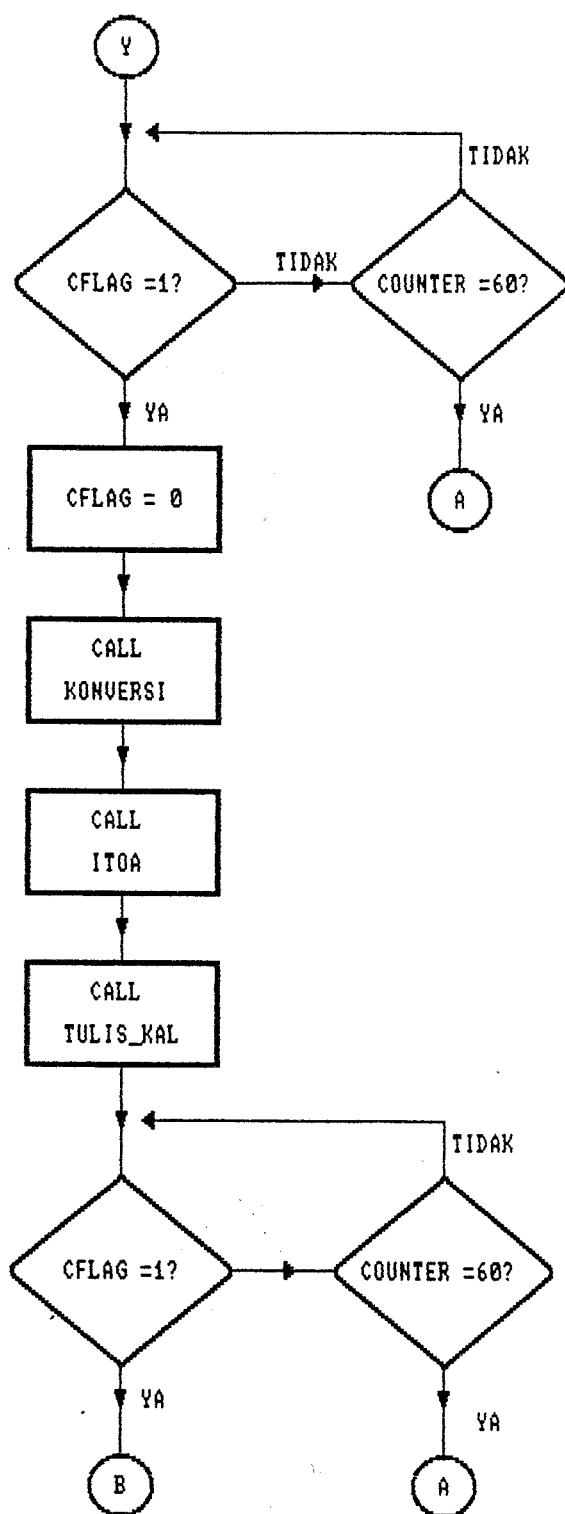
Title ADC 8884

Size	Number	Revision
A4		2
Date: 22-APR-1994 Sheet 1 of 4		
File: D:\PRITHA\SKEMA\AD Drawn By: PRITA		





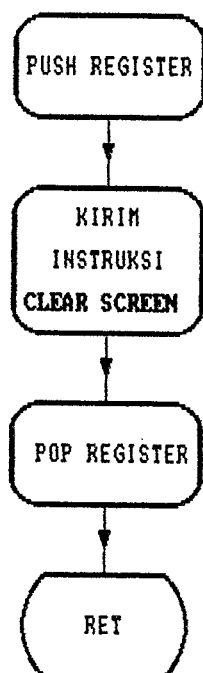




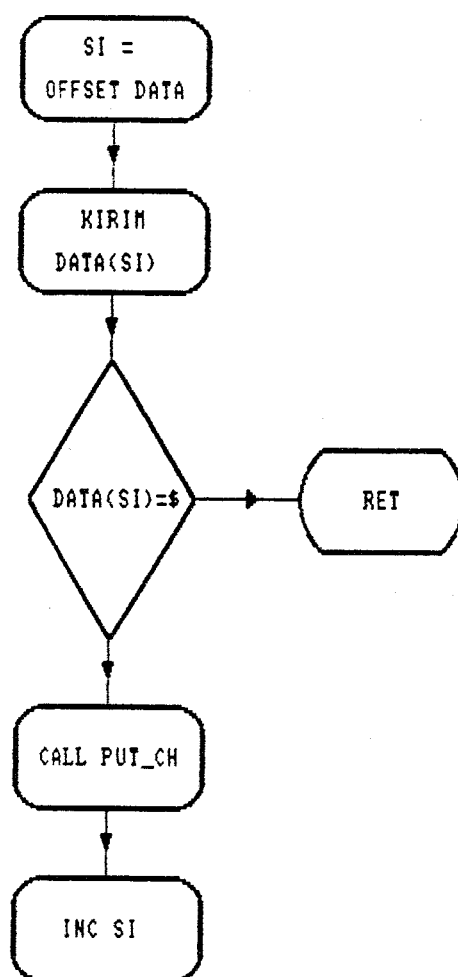
PROCEDURE
RET_HOME



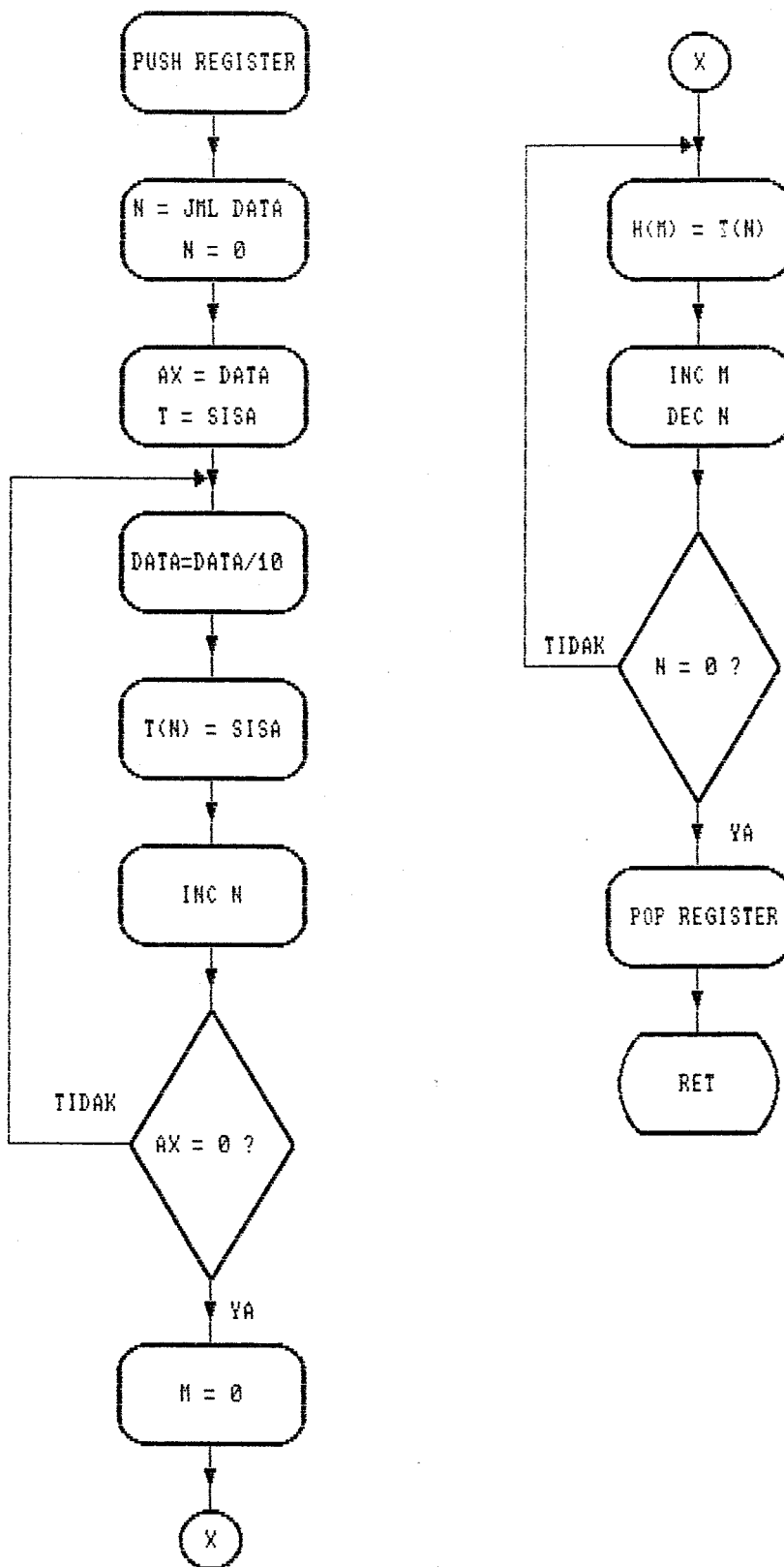
PROCEDURE
CLR_SCR



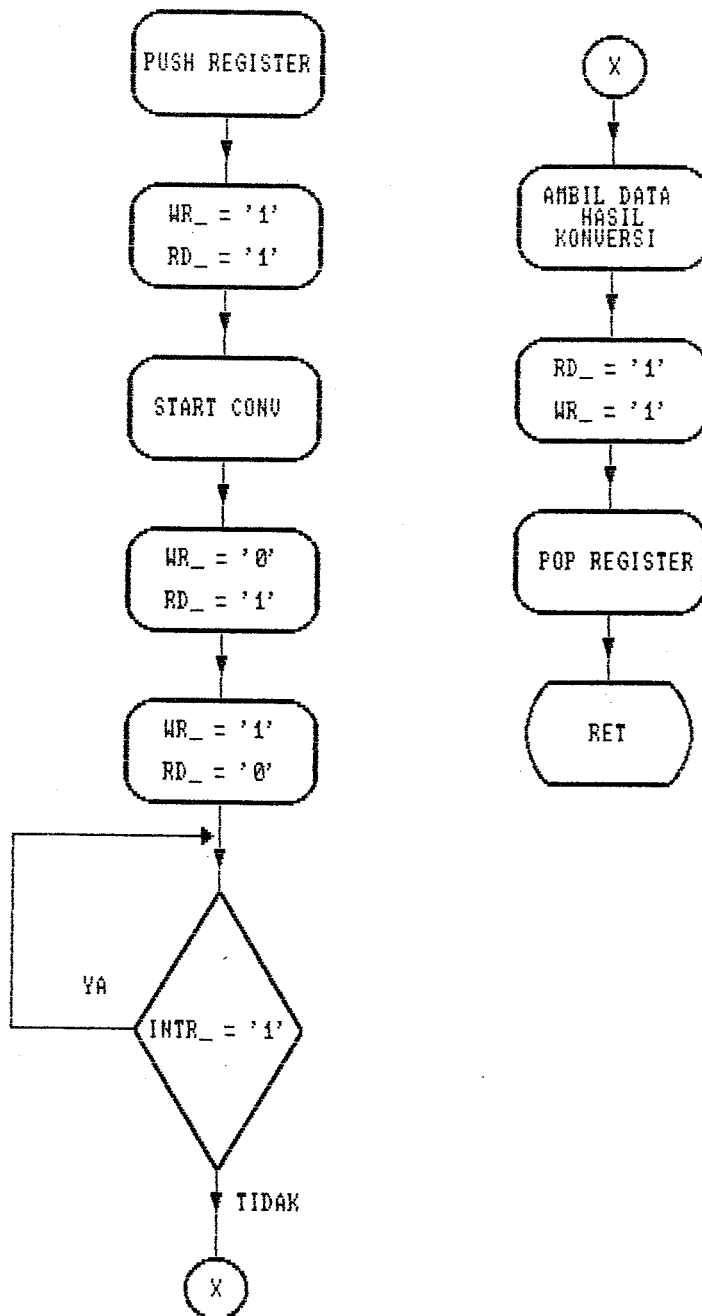
PROCEDURE
TULIS_KAL



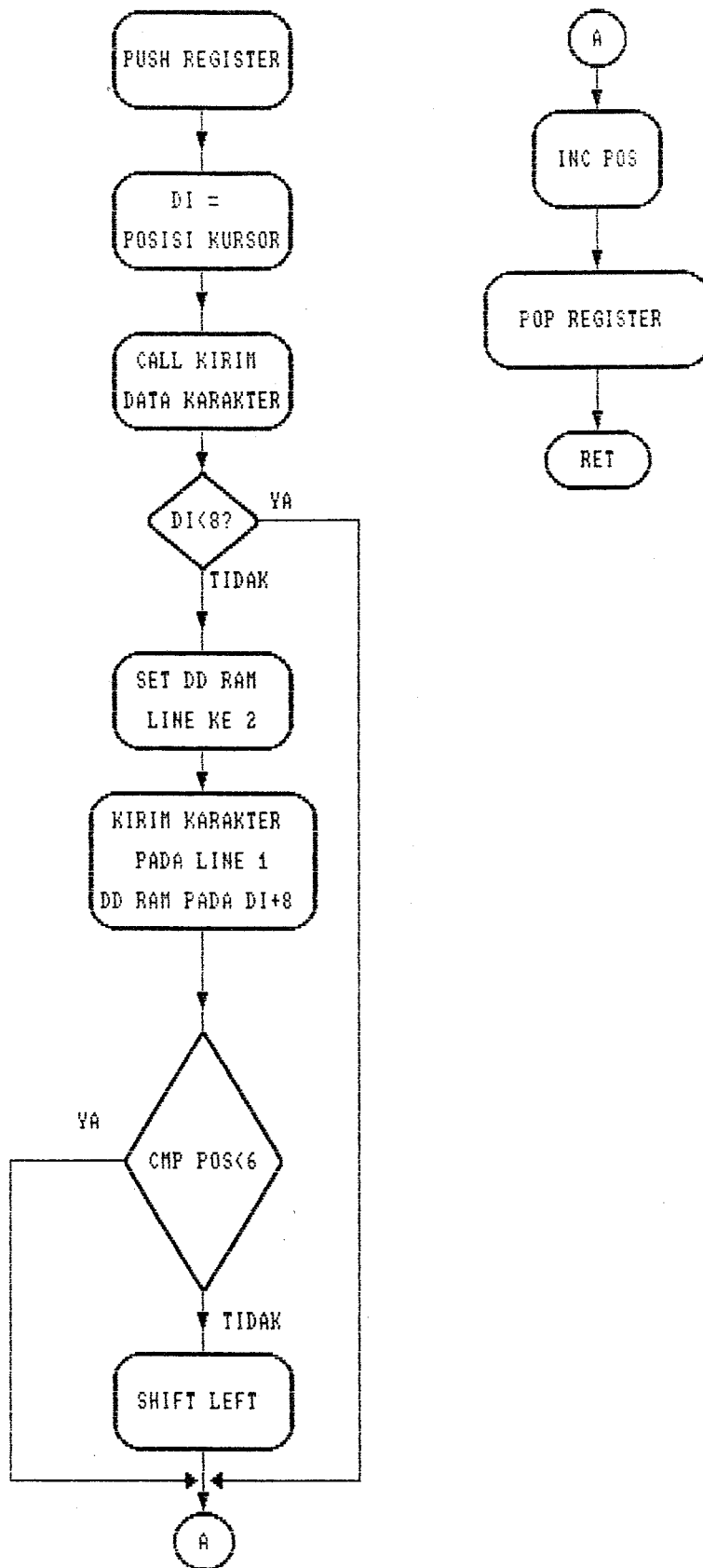
PROCEDURE KONVERSI INTEGER KE ASCII
(PROC ITOA)

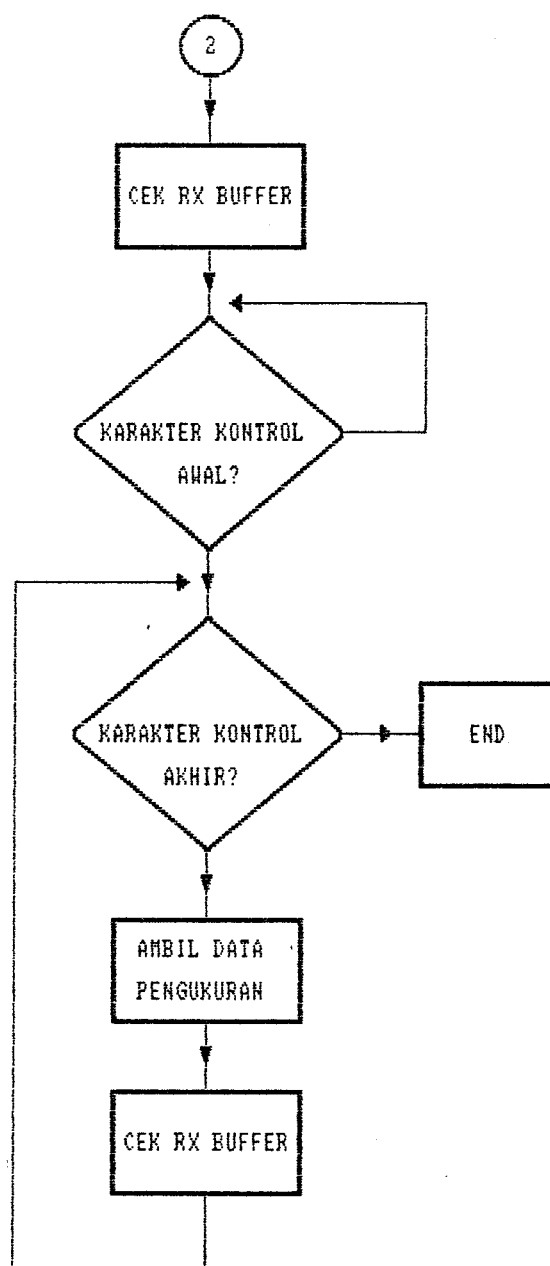
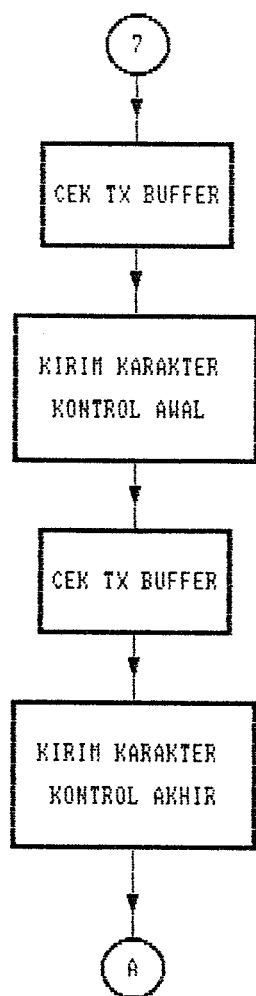


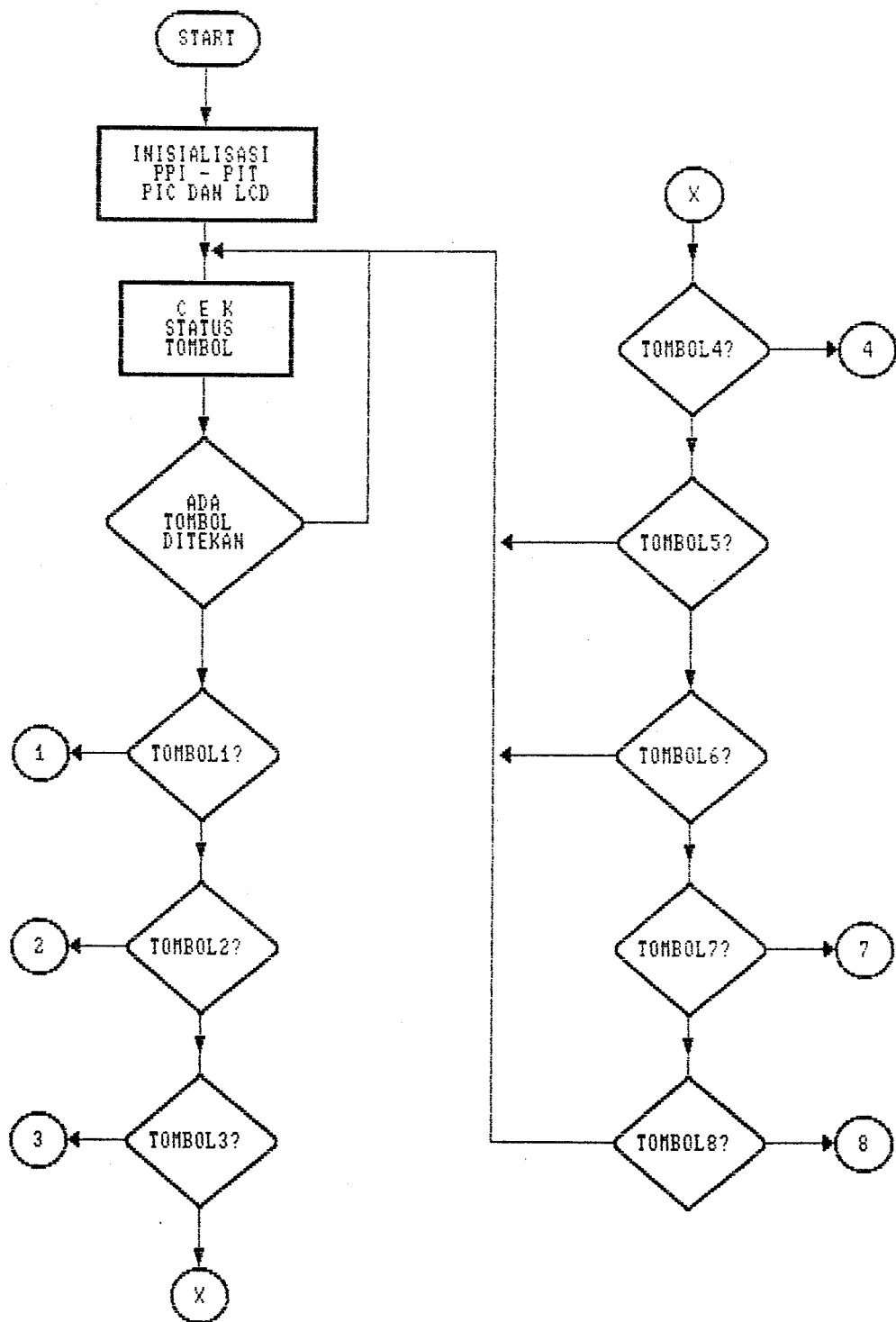
PROCEDURE KONVERSI ANALOG KE DIGITAL
(PROC KONVERSI)



PROCEDURE PUT_CH







port_a0 equ 000h
port_b0 equ 001h
port_c0 equ 002h
port_cw0 equ 003h

port_a equ 30Ch
port_b equ 30Dh
port_c equ 30Eh
port_cw equ 30Fh

timer_0 equ 004h
timer_1 equ 005h
timer_2 equ 006h
timer_cw equ 007h

INTA00 equ 008h
INTA01 equ 009h

TX_BUFFER EQU 03F8H
RX_BUFFER EQU 03F8H
BAUD_DIV_LSB EQU 03F8H
BAUD_DIV_MSB EQU 03F9H
INT_ENB_REG EQU 03F9H
INT_ID_REG EQU 03FAH
LINE_CONT_REG EQU 03FBH
MODEM_CONT_REG EQU 03FCH
LINE_STAT_REG EQU 03FDH
MODEM_STAT_REG EQU 03FEH

DATAAREA EQU 10h

start1:

-----;
; Data-data RAM
-----;

_DATA SEGMENT AT 10h
ORG 0h

pos dw ?
temp db 8 dup(?)
t db ?
mux db ?
hasil db 8 dup(?)
count db ?
cflag db ?
counter db ?
posisi dw ?
minus db ?
nilai db 8 dup(?)
jml dw ?
jum dw ?
status db ?
out_adc db 100 dup(?)
out_adcl db 100 dup(?)
tinggi db ?

```

tinggil db ?
conv_bcd db 2 dup(?)
_DATA   ENDS

```

```

;-----;
; Program pada EPROM ;
;-----;

```

```

_TEXT   SEGMENT BYTE
        ASSUME CS:_TEXT,DS:_DATA
        ORG 100h

```

```

start:

```

```

;-----;
; inisialisasi segment register ;
;-----;

```

```

        cli
        mov ax,DATAAREA
        mov ds,ax

        xor ax,ax
        mov es,ax
        mov ss,ax
        mov sp,800h

        mov bx,4*08h      ; load vector table
        mov ax,offset timer_int ; load type 8
        mov es:[bx],ax
        mov ax,cs
        mov es:[bx+2],ax

        mov ax,DATAAREA
        mov es,ax

```

```

;-----;
; inisialisasi PPI ;
;-----;

```

```

        mov al,80h      ; PPI untuk LCD
        mov dx,port_cw
        out dx,al

        mov al,10011010b ; PPI untuk ADC dan keypad
        mov dx,port_cw0
        out dx,al

```

```

;-----;
; Inisialisasi timer ;
;-----;

```

```

        mov al,00111110b
        out timer_cw,al

```

```

mov al,03Ah          ; mode 3, pembagi 14906
out timer_0,al       ; 20 Hz, msb dulu baru lsb

```

```

mov al,03Ah
out timer_0,al

```

```

mov counter,0
mov count,0
mov cflag,0

```

```

;-----;
;inisialisasi PIC      ;
;-----;

```

```

mov al,13h           ;ICW1 - LTIM = 0,S=1,IC4=1
out INTA00,al

mov al,08h           ;ICW2-INTERRUPT type 8 (08h)
out INTA01,al

mov al,09h           ;ICW4-SFNM=0,BUFF=1,AEOI=0,MPN=1
out INTA01,al

mov al,1111110b      ;OCW1-MASK IR 2-7 (not used)
out INTA01,al

sti

```

```

;-----;
;inisialisasi LCD      ;
;-----;

```

```

mov al,0h
mov dx,port_b
out dx,al

mov cx,0
call delay           ; tunggu power on

mov al,00111000b     ; function set
mov cx,4000
call kirim_inst

mov cx,100
call kirim_inst

mov al,00111000b     ; function set
mov cx,115
call kirim_inst

mov al,00001000b     ; display off
mov cx,115
call kirim_inst

```



```

mov al,00000001b      ; clear display
mov cx,115
call kirim_inst

mov al,00000110b      ; entry mode set
mov cx,115
call kirim_inst

mov al,00001100b      ; display on, cursor tidak tampak
mov cx,115
call kirim_inst
call clrscr
mov cx,0
call delay

```

```

;-----;
;inisialisasi serial RS-232C;
;-----;

```

```

mov dx,line_cont_reg
mov al,1000000b
out dx,al

mov dx,baud_div_msb
mov al,01h
out dx,al

mov dx,baud_div_lsb
mov al,80h      ; baud rate 300
out dx,al

mov dx,line_cont_reg
mov al,00000011b      ; 8 data bit, 1 stop bit, parity disable
out dx,al

mov dx,int_enb_reg
mov al,00h
out dx,al

```

```

;=====;
;          M A I N      P R O G R A M          ;
;=====;

```

```

mulai:
mov dx,port_b0
nokey:  in  al,dx
        push cx      ; mendeteksi bouncing
        mov cx,16eah
del:    loop del
        pop cx
        cmp al,0FFh   ; jika al = 0FFh maka
        je  nokey     ; tidak ada tombol ditekan

```

```

        mov  posisi,0
scan:    shr  al,1          ; Cari posisi tombol
        jnc  yes
        inc  posisi
        cmp  posisi,8
        jb   scan
        jmp  nokey

```

```

yes:     cmp  posisi,0
        je   awal
        cmp  posisi,1
        je   kalibrasi
        cmp  posisi,2
        je   gigi_seri
        cmp  posisi,3
        je   geraham
        cmp  posisi,4
        je   nokey
        cmp  posisi,5
        je   nokey
        cmp  posisi,6
        je   clear
        cmp  posisi,7
        je   kirim_data
        jmp  nokey

```

```

;-----;
; Tombol START
;-----;

```

```

awal:    mov  pos,0
        call clrscr
        push cs
        pop  es
        mov  si,offset ready
        call tulis_kal
        call ret_home
        jmp  mulai

```

```

;-----;
;Kalibrasi ADC
;-----;

```

```

kalibrasi: mov  pos,0
        jmp  mulai
        mov  minus,0
        call konversi
        mov  minus,al
        jmp  mulai

```

```

geraham:  jmp  geraham1
clear:    jmp  clear1
kirim_data: jmp  kirim_data1

```

```

;-----;
;Pengukuran pada Gigi Seri;
;-----;

```

gigi_seri:

```

    mov status,0
    mov mux,0h          ; input 1 aktif
    mov pos,0
    mov counter,0

```

```

mux0:    mov di,offset out_adc+1
    mov cflag,0
    call konversi
    mov byte ptr [di],al
    inc di

```

```

cek:     cmp cflag,1
    je mux0
    cmp counter,60      ; 3 detik
    jb cek
    mov jum,di

```

```

again:   dec di
    cmp di,offset out_adc
    je go
    mov al,byte ptr [di]
    cmp al,byte ptr [di+1]
    jb again
    mov al,byte ptr [di+1]
    jmp again

```

go:

```

    xor ah,ah
    mov bl,195
    mul bl
    mov cx,ax           ; setelah semua selesai
    mov di,offset hasil ; tampilkan data tertinggi
    call itoa
    call clrscr
    push ds
    pop es
    mov si,offset hasil
    call tulis_kal
    call ret_home

```

```

;    call kalib
    jmp mulai

```

```

;-----;
;Pengukuran pada Gigi Geraham;
;-----;

```

```

geraham1: mov status,1
    mov counter,0

```

```

        mov di,offset out_adc+1 ; geraham kanan
        mov si,offset out_adc1+1 ; geraham kiri

mux1:    mov cflag,0
        mov mux,00000001b      ; input 2 aktif, geraham kanan
        mov pos,0
        call konversi
        mov byte ptr [di],al
        inc di
        mov mux,00000010b      ; input 3 aktif, geraham kiri
mux2:    call konversi
        mov byte ptr [si],al
        inc si

cek1:    cmp cflag,1
        je mux1
        cmp counter,60         ; 3 detik
        jb cek1
        mov jml,di
        mov jum,si

again2:  dec di
        cmp di,offset out_adc
        je next1
        mov al,byte ptr [di]
        cmp al,byte ptr [di+1]
        jb again2
        mov bl,byte ptr [di+1]
        jmp again2

next1:   dec si
        cmp si,offset out_adc1
        je gol
        mov al,byte ptr [si]
        cmp al,byte ptr [si+1]
        jb next1
        mov dl,byte ptr [si+1]
        jmp next1

gol:     xor ah,ah
        mov al,bl
        mov bl,195
        mul bl
        mov cx,ax              ; setelah semua selesai
        mov di,offset hasil    ; tampilkan data tertinggi
        call itoa
        call clrscr
        push ds
        pop es
        mov si,offset hasil
        call tulis_kal
        call ret_home

go2:     xor ah,ah
        mov al,dl

```

```

    mov bl,195
    mul bl
    mov cx,ax          ; setelah semua selesai
    mov di,offset hasil ; tampilkan data tertinggi
    call itoa
    call line2
    push ds
    pop es
    mov si,offset hasil
    call tulis_kali
    call ret_home

    call kalib
    jmp mulai

```

```

;-----;
;pengiriman data serial ;
;status = 0, pengukuran gigiseri ;
;status = 1, pengukuran geraham ;
; * reg [di] genap -> input 2 ;
; * reg [di] ganjil -> input 3 ;
;-----;

```

```

kirim_data1: call cek_tx
              cmp status,1
              je hasil_ger

```

```

lagil:      mov si,offset out_adc
            mov dx,tx_buffer
            mov al,12h          ; kontrol karakter awal untuk gigiseri
            out dx,al
            call cek_tx

```

```

digit:      xor ah,ah
            mov al,byte ptr [si]
            mov cx,ax
            mov di,offset hasil
            call itoa
            mov di, offset hasil
            mov al,byte ptr [di]
            out dx,al
            call cek_tx
            inc di
            mov bx,06
            add bx,offset hasil
            cmp di,bx
            jb digit

```

```

            mov al,13h          ; kontrol karakter akhir
            out dx,al
            call cek_tx

```

```

            inc si
            cmp si,jum
            jne lagil
            jmp mulai

```

```

hasil_ger:  mov si,offset out_adc
lagi2:      mov dx,tx_buffer
            mov al,10h          ; kontrol karakter awal geraham
            out dx,al
            call cek_tx

digit1:     xor ah,ah
            mov al,byte ptr [si]
            mov cx,ax
            mov di,offset hasil
            call itoa
            mov di, offset hasil
            mov al,byte ptr [di]
            out dx,al
            call cek_tx
            inc di
            mov bx,6
            add bx,offset hasil
            cmp di,bx
            jb digit1
            mov al,11h          ; kontrol karakter akhir
            out dx,al
            call cek_tx
            inc si
            cmp si,jml
            jne lagi2

            jmp mulai

```

```

;-----
;clear memory
;-----

```

```

clear1:     mov pos,0
            call clrscr
            push cs
            pop es
            mov si,offset clear
            call tulis_kal
            call ret_home
            jmp start1

```

```

;=====;
;          P R O C E D U R E      P R O G R A M          ;
;=====;

```

```

;-----;
;kalibrasi ;
;(output adc saat tanpa input) ;
;disimpan pada register minus ;
;-----;

```

```

kalib:      push ax
            call konversi
            mov minus,al
            pop ax
            ret

```

```
; DISPLAY SERVICE
```

```
;-----;  
; Mengembalikan kursor pada posisi awal ;  
;-----;
```

```
ret_home PROC
```

```
    push ax
```

```
    mov pos,0
```

```
    mov al,00000010b      ; return home
```

```
    mov dx,port_a
```

```
    out dx,al
```

```
    call instruksi
```

```
    mov cx,200
```

```
    call delay
```

```
    pop ax
```

```
    ret
```

```
ret_home ENDP
```

```
;-----;  
; Menghapus layar  
;-----;
```

```
clrscr PROC
```

```
    push ax
```

```
    push di
```

```
    mov pos,0
```

```
    mov al,00000001b      ; clear display
```

```
    mov dx,port_a
```

```
    out dx,al
```

```
    call instruksi
```

```
    mov cx,500
```

```
    call delay
```

```
    pop di
```

```
    pop ax
```

```
    ret
```

```
clrscr ENDP
```

```
;-----;  
; prosedure delay  
; delay = cx * 5 * 210 ns  
;-----;
```

```
delay proc
```

```
    loop $
```

```
    ret
```

```
delay endp
```

```

;-----;
;procedure untuk pengiriman data ;
;-----;

```

```

tulis    proc
          push ax
          mov  al,01000001b      ; RS high, enable
          mov  dx,port_b
          out  dx,al
          mov  cx,1
          call delay
          mov  al,00000001b      ; RS high, disable
          mov  dx,port_b
          out  dx,al
          mov  cx,1
          call delay
          pop  ax
          ret
tulis    endp

```

```

;-----;
;procedure untuk pengiriman instruksi ;
;-----;

```

```

instruksi PROC
          push ax
          mov  al,01000000b      ; RS low, enable
          mov  dx,port_b
          out  dx,al
          mov  cx,1
          call delay
          mov  al,00000000b      ; RS low, disable
          mov  dx,port_b
          out  dx,al
          mov  cx,1
          call delay
          pop  ax
          ret
instruksi endp

```

```

;-----;
; Kirim satu instruksi ;
; register al = instruksi, register cx = waktu delay ;
;-----;

```

```

kirim_inst PROC
          push ax
          push cx
          mov  dx,port_a
          out  dx,al
          call instruksi
          call delay
          pop  cx
          pop  ax
          ret
kirim_inst ENDP

```



```

    kirim_satu proc
        push ax
        mov dx,port_a
        out dx,al
        call tulis
        mov cx,100
        call delay
        pop ax
        ret
    kirim_satu endp

```

```

;-----;
; tulis kalimat pada posisi kursor ;
; register bl = offset alamat kalimat ASCII ;
;-----;

```

```

put_ch    PROC
    push ax
    push bx
    push di

    mov di,pos

@norm:    mov al,bl

    call kirim_satu
    cmp di,8
    jb @line1

    mov ax,di
    mov ah,11000000b    ; set DD RAM 40h
    add al,ah
    sub al,8
    mov cx,1
    call kirim_inst

    mov al,bl
    call kirim_satu

    mov ax,di
    mov ah,10000000b    ; set DD RAM 00h
    add al,ah
    inc al
    mov cx,1
    call kirim_inst

    cmp pos,16
    jb @line1

    mov al,00011000b    ; shift left
    mov cx,1
    call kirim_inst

@line1:   inc pos

```

@akhir:

```
pop di
pop bx
pop ax
ret
```

put_ch ENDP

```
;-----;
; Set alamat DD RAM 40h
;-----;
```

```
line2 PROC
push ax
mov al,11000000b ; set DD RAM 40h
mov cx,1
call kirim_inst
pop ax
ret
line2 ENDP
```

```
;-----;
;penulisan kalimat
;-----;
```

```
tulis_kal proc
push ax
push bx
push cx
ul: mov al,es:[si]
cmp al,'$'
je selesai
mov bl,al
call put_ch
inc si
mov cx,100
call delay
jmp ul
selesai: pop cx
pop bx
pop ax
ret
tulis_kal endp
```

```
put_ch1 PROC
push ax
push bx
push di

mov di,pos
```

```
@norm1: mov al,bl
call kirim_satu

mov ax,di
mov ah,11000000b ; set DD RAM 40h
add al,ah
```

```

        mov cx,1
        call kirim_inst

        inc pos
@akhiri:
        pop di
        pop bx
        pop ax
        ret
put_ch1  ENDP

tulis_kali proc
        push ax
        push bx
        push cx
        mov pos,1
ul1:     mov al,es:[si]
        cmp al,'$'
        je  selesai
        mov bl,al
        call put_ch1
        inc si
        mov cx,100
        call delay
        jmp ul
selesai: pop cx
        pop bx
        pop ax
        ret
tulis_kali endp

tulis_dgt proc
        push ax
        push bx
        push cx
        mov pos,0
        call line2
ulg:     mov al,es:[si]
        cmp al,'$'
        je  finish
        mov bl,al
        call put_ch
        inc si
        mov cx,100
        call delay
        jmp ulg

finish:  pop cx
        pop bx
        pop ax
        ret
tulis_dgt endp

; END DISPLAY SERVICE

```

```

;-----;
; Konversi integer ke ASCIIIZ ;
; register cx = harga integer, di = Hasil ASCIIIZ ;
;-----;
Itoa      proc near
            push ax
            push bx
            push cx
            push si
            push di

            mov si,offset temp+1

@lagi:
            mov ax,cx
            mov bx,10
            xor dx,dx
            idiv bx
            mov t,dl

            mov cx,ax

            mov al,t
            add al,30h
            mov byte ptr [si],al

            inc si

            or cx,cx
            jne short @lagi

            mov ax,si
            mov bx,offset temp+1
            sub ax,bx
            mov bx,6
            sub bx,ax

blank:     mov byte ptr [si], 30h
            inc si
            dec bx
            or bx,bx
            jne blank

            dec si

            jmp short @ckptr

@nextdgt:
            mov al,byte ptr [si]
            mov byte ptr [di],al
            dec si
            inc di

@ckptr:
            cmp si,offset temp
            jne short @nextdgt

            mov byte ptr [di],'$' ; tanda akhir data

```

```

        pop di
        pop si
        pop cx
        pop bx
        pop ax
        ret
Itoa    endp

```

```

;-----;
; Konversi ADC
; register al = hasil konversi
;-----;

```

```

konversi proc
    push bx
    push dx
    mov dx,port_c0
    mov al,00001100b    ; WR_ '1',
    add al,mux
    out dx,al

    mov dx,port_c0
    mov al,00001000b    ; start conversion, PC2 -> WR_, PC3 -> RD_
    add al,mux
    out dx,al
    mov cx,2
    call delay

    mov al,00001100b
    add al,mux
    out dx,al
    mov cx,2
    call delay

    mov dx,port_c0
lagi:   in al,dx
    test al,00010000b    ; PC4 -> INT_
    jnz lagi

    mov dx,port_c0
    mov al,00000100b
    add al,mux
    out dx,al

    mov dx,port_a0
    in al,dx
    sub al,minus
    mov bl,al

    mov dx,port_c0
    mov al,00001100b
    add al,mux
    out dx,al

```

```

        xor ah,ah
        mov al,bl
        pop dx
        pop bx
        ret
konversi endp

```

```

;-----;
; Timer service
;-----;

```

```

timer_int proc near
    cli
    push ax
    push bx
    push cx
    push dx
    push si
    push di
    push es
    push ds

```

```

    inc counter
    inc count
    cmp count,5
    jb T1
    mov count,0
    mov cflag,1

```

```

T1:     mov ax,20h
        mov dx,INTA00
        out dx,al

```

```

        pop ds
        pop es
        pop di
        pop si
        pop dx
        pop cx
        pop bx
        pop ax
        sti
        iret

```

```

timer_int endp

```

```

;-----;
;pengontrolan Tx buffer pada
;pengiriman data serial
;-----;

```

```

cek_tx  proc
        push ax
        push dx
cektx:   mov dx,line_stat_reg
        in  al,dx
        cmp al,00100000b      ; cek apakah register Tx kosong

```

```
        jne cektx
        pop dx
        pop ax
        ret
cek_tx  endp
```

```
-----;
; Definisi data tetap;
-----;
```

```
nama    db 'Dyah Patria Nurhayati -- Elektro ITS ','$'
ready   db 'READY ','$'
clear   db 'CLEAR ','$'
```

```
_TEXT  ENDS
        END start
```